Discrete Mathematics and Theoretical Computer Science DMTCS vol. **16:3**, 2014, 23–50

A Four-Sweep LBFS Recognition Algorithm for Interval Graphs[*]

Peng Li[†] and Yaokun Wu[‡]

Department of Mathematics, and MOE-LSC, Shanghai Jiao Tong University, Shanghai, 200240, China

received 4th Dec. 2011, revised 17th Jan. 2014, 31st May 2014, accepted 11th June 2014.

In their 2009 paper, Corneil *et al.* design a linear time interval graph recognition algorithm based on six sweeps of Lexicographic Breadth-First Search (LBFS) and prove its correctness. They believe that their corresponding 5-sweep LBFS interval graph recognition algorithm is also correct. Thanks to the LBFS structure theory established mainly by Corneil *et al.*, we are able to present a 4-sweep LBFS algorithm which determines whether or not the input graph is a unit interval graph or an interval graph. Like the algorithm of Corneil *et al.*, our algorithm does not involve any complicated data structure and can be executed in linear time.

Keywords: chordal graph, interval graph, interval representation, Lexicographic Breadth-First Search, perfect ordering, recognition algorithm.

1 Introduction

In this paper, all graphs are assumed to be finite, simple, undirected and loopless. For any graph G, a map $\mathcal{I}$ that assigns to each vertex $x \in V(G)$ a nonempty closed interval $\mathcal{I}(x) = [\ell_{\mathcal{I}}(x), r_{\mathcal{I}}(x)]$ is called an *interval representation* of G provided $xy \in E(G)$ if and only if $x \neq y$ and $\mathcal{I}(x) \cap \mathcal{I}(y) \neq \emptyset$ for all $x, y \in V(G)$. If $r_{\mathcal{I}} - \ell_{\mathcal{I}}$ takes a constant value, we refer to the interval representation $\mathcal{I}$ as a *unit interval representation*. A graph is an *interval graph* if and only if it has an interval representation and a graph is a *unit interval graph* if and only if it has a unit interval representation. Interval graphs arise so naturally and frequently in modeling real-life situations that they must be studied from the application viewpoint. On the other hand, the study of interval graphs has displayed intrinsic beauty and interest which attract much attention of mathematicians simply due to its elegance [18, 20, 41, 49].

For any two integers i and j we write $[i, j]$ for the set of integers k such that $i \leq k \leq j$. For a map σ defined on $[i, j]$, $\sigma[i, j]$ stands for $\{\sigma(k) : \ k \in [i, j]\}$. For any integer j we will often write $[j]$ for $[1, j]$ and hence $\sigma[j]$ for $\sigma[1, j]$.

[*]This work was supported by the National Natural Science Foundation of China (No. 11271255). The authors gratefully acknowledge the many helpful suggestions from the anonymous referees.

[†]Email: lidetiansjtu@sjtu.edu.cn

[‡]Corresponding author, email: ykwu@sjtu.edu.cn

1365–8050 © 2014 Discrete Mathematics and Theoretical Computer Science (DMTCS), Nancy, France

Let G be a graph and let $x \in V(G)$. The *open neighborhood* of x in G, denoted by $N_G(x)$, is the set of vertices adjacent to x in G. The *closed neighborhood* of x in G, denoted by $N_G[x]$, is $N_G(x) \cup \{x\}$. If $x_0, \ldots, x_\ell$ is a sequence of distinct vertices such that $x_{i-1}x_i \in E(G)$ for every $i \in [\ell]$, we call it an x_0, x_ℓ*-path of length* ℓ. An *asteroidal triple* (or *AT* for short) of a graph is a set of three vertices such that every pair of vertices are joined by a path outside of the closed neighborhood of the third. A graph is *AT-free* if it contains no asteroidal triple. A graph is *chordal* if it contains no induced cycle of length greater than 3.

Theorem 1.1 *[34] A graph is an interval graph if and only if it is chordal and AT-free.*

Let G be a graph and let a, b, c, d be four vertices in G. We say that $\{a, b, c, d\}$ is a claw centered at a in G provided b, c and d are three independent vertices from $N_G(a)$.

Theorem 1.2 *[45] An interval graph is a unit interval graph if and only if it is claw-free.*

Let S be a set of size n. An *ordering* of S is a bijective map from $[n]$ to S. Let G be a graph on n vertices and σ an ordering of $V(G)$. We call σ an *I-ordering of* G if for all $i, k \in [n]$ such that $\sigma(i)\sigma(k) \in E(G)$ it holds $\sigma(i)\sigma(j) \in E(G)$ for every $j \in [i+1, k-1]$. We say that σ is a *UI-ordering of* G provided $\{\sigma(i)\sigma(j), \sigma(j)\sigma(k)\} \subseteq E(G)$ for all i, j, k satisfying $j \in [i+1, k-1]$ and $\sigma(i)\sigma(k) \in E(G)$.

Theorem 1.3 *[43, 44] A graph is an interval graph if and only if it has an I-ordering.*

Theorem 1.4 *[37] A graph is a unit interval graph if and only if it has a UI-ordering.*

Theorems 1.1 and 1.2 provide negative certificates for interval graphs and unit interval graphs respectively while Theorems 1.3 and 1.4 suggest positive certificates for them. Based on these and other characterizations, many linear time recognition algorithms of interval graphs and unit interval graphs have been developed [1, 2, 4, 6, 11, 13, 14, 15, 17, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 35, 38, 40, 42]. However, among all of these many linear time algorithms, it is argued that those based on the so-called Lexicographic Breadth-First Search involve relatively simple data structure and are hence easier to implement [5, 11, 13, 48].

Lexicographic Breadth-First Search, often abbreviated as `LBFS`, is a linear time algorithm for ordering the vertex set of a given graph and is developed by Rose, Tarjan and Lueker in 1976 [5, 13, 46]. By introducing some tie-breaking rules, one can restrict the range of the output orderings of the generic `LBFS` and get better orderings of the vertex set for the purpose of a specific algorithm. Some special tie-breaking rules have been suggested due to various algorithm design considerations and so we can find from literature several variants of the generic `LBFS`. A widely used one is the `LBFS+` algorithm discovered by Ma [39] and Simon [47] independently, which breaks tie according to a given vertex ordering of the input graph. In an unpublished manuscript, Corneil and Köhler [7] show that $O(n)$ sweeps of `LBFS+` will produce an I-ordering as a positive certificate for a given interval graph on n vertices and $o(n)$ sweeps will not suffice. Corneil *et al.* [11] suggest a variant of `LBFS` called `LBFS`*. To break tie, it makes use of the information from the previous two vertex orderings. In [11, 13], Corneil *et al.* establish a huge theory of the `LBFS` properties of interval graphs, based on which they employ six passes of `LBFS`, which consists of one generic `LBFS`, then four sweeps of `LBFS+` and finally a sweep of `LBFS`*, to get a simple recognition algorithm for general interval graphs. They [11] once think that the four middle sweeps of `LBFS+` can be replaced by only two such sweeps, which turns out to be false. However, they [13] still believe that one can substitute three sweeps of `LBFS+` for the four middle sweeps and mention that maybe an even more complicated argument will be needed to prove the correctness of the new algorithm. For any

unit interval graph, Corneil [4] finds that three sweeps of LBFS+ are enough to construct a UI-ordering of it. We recently develop this work of Corneil in [35] by inventing a new variant of LBFS and a byproduct of this effort is a linear time 2-sweep LBFS certifying algorithm for recognizing unit interval graphs [35, §2].

Motivated by the above line of work, this paper will further study the LBFS properties of interval graphs and propose a linear time 4-sweep LBFS algorithm for recognizing interval graphs. Note that each sweep in our algorithm only makes use of the information of the previous sweep, unlike the LBFS* sweep used by Corneil *et al.* Based on our earlier work on recognizing unit interval graphs [35], we will show that applying our interval graph recognition algorithm on a unit interval graph will result in a UI ordering and hence our algorithm is simultaneously a unit interval graph recognition algorithm. By contrast, it is not clear whether or not the 6-sweep LBFS algorithm of Corneil *et al.* [13] can recognize unit interval graphs. We remark that our paper also suggests a 3-sweep LBFS algorithm for recognizing interval graphs, though we do not see any linear time implementation of it.

Till now, seems that the only known linear time certifying recognition algorithm for interval graphs is reported by Kratsch *et al.* [33]. This algorithm makes use of the recognition algorithm of Korte and Möhring [32], which in turn applies the data structure called MPQ tree. We mention that, after adding an additional certifying step, our 4-sweep LBFS interval graph recognition algorithm can be further developed into a certifying algorithm without any usage of a data structure like MPQ tree and we will present this certifying algorithm as well as a proof of its correctness in another paper [36]. Indeed, some results in this paper will be reproved in [36] with longer constructive arguments as we need to show how to catch the negative certificates efficiently.

The remainder of the paper is organized as follows. Section 2 is a brief introduction of some basic LBFS algorithms with different rules of breaking ties. In Section 3, we present our 4-sweep LBFS interval graph recognition algorithm. Section 4 is the main part of this paper in which we collect various theoretical and algorithmic results concerning LBFS and interval representations and can thus demonstrate there the correctness of our algorithm. Note that some lemmas in Section 4 are formulated in a general form so that we can directly use them in the forthcoming paper [36] on analyzing the corresponding certifying algorithm. Finally, we present a linear time implementation of our algorithm in Section 5.

2 LBFS orderings

For two different sets S and T of positive integers, we say that S is *lexicographically smaller* than T if the minimum element from the symmetric difference of S and T comes from T. We warn the readers that many authors use "lexicographically bigger" for what we call here "lexicographically smaller".

Let σ be an ordering of S and let T be a subset of S of size k. The *ordering of T induced by σ* is a map σ^T from $[k]$ to T such that $\sigma^T(i) = \sigma(s_i)$ for $i \in [k]$, where $s_1 < \cdots < s_k$ and $T = \{\sigma(s_1), \ldots, \sigma(s_k)\}$.

Let G be a graph on n vertices and let σ be an ordering of $V(G)$. We say that $\sigma(i)$ appears *earlier* than $\sigma(j)$ if $1 \leq i < j \leq n$ and we call $\sigma(n)$ the *last* vertex in the ordering σ. For any two different subsets S and T of $V(G)$, we write $S <_\sigma T$ when $\{i : \sigma(i) \in S\}$ is lexicographically smaller than $\{i : \sigma(i) \in T\}$. We adopt the notation $S \leq_\sigma T$ when $S = T$ or $S <_\sigma T$. Take $k \in [n]$ and $w \in V(G)$. We use $L^k_{G,\sigma}(w)$ to denote the set $N_G[w] \cap \sigma[k]$ and we regard $L^0_{G,\sigma}(w)$ as the empty set. For any $x, y \in \sigma[k, n]$, we write $x <_{\sigma,k-1} y$ provided $L^{k-1}_{G,\sigma}(x) <_\sigma L^{k-1}_{G,\sigma}(y)$. The LBFS *$\sigma$-slice in G at time k*, which is denoted by $S_{\sigma,G}(k)$, or simply $S_\sigma(k)$ if the graph G is clear from context, is the set of maximal elements for the partial order $<_{\sigma,k-1}$. Note that the slice $S_\sigma(k)$ is already determined by the ordering $\sigma^{[k-1]}$ and

the structure of the graph G and so sometimes we will use the same notation $S_\sigma(k)$ when we know the definition of σ on $[k-1]$ and do not care about whether or not σ is defined on $[k,n]$. For brevity, we often refer to an LBFS slice simply as a slice. A slice is *nontrivial* if it contains at least two elements.

For the purpose of this paper, we first give a description of the generic LBFS which starts from a distinguished vertex but then allows arbitrary tie-breaking later on.

```
Procedure LBFS(G, u);
{ Input: a graph G and a vertex u ∈ V(G);
Output: an ordering τ of V(G)}
begin
    label(u) ← {1};
    for each vertex v in V(G)\{u} do
        label(v) ← ∅;
    for i ← 1 to |V(G)| do begin
        pick an unvisited vertex v with lexicographically the largest label; (♣)
        τ(i) ← v;
        for each unvisited vertex w in N_G(v) do
            label(w) = label(w) ∪ {i};
        end
    end; {LBFS}
```

Let G be a graph and take $u \in V(G)$. An LBFS *ordering of G beginning at $u \in V(G)$* is any output of the algorithm LBFS(G,u). In general, an LBFS *ordering of G* is any LBFS ordering of G beginning at some vertex of G. It is easy to see that an ordering σ of an n-vertex graph is an LBFS ordering if and only if $\sigma(k) \in S_\sigma(k)$ for every $k \in [n]$. Note that the LBFS orderings are characterized by the so-called property (P1) [16, 31].

For any graph G and any $x, y \in V(G)$, we write $d_G(x,y)$ for the length of a shortest path from x to y in G.

Remark 2.1 *Let G be a graph on n vertices and let ρ be any breadth-first search (BFS) ordering, say an* LBFS *ordering, of G. Then $d_G(\rho(i),\rho(1))$ is nondecreasing for $i \in [n]$.* □

By introducing some tie-breaking rule, Ma [39] and Simon [47] independently propose the following very useful LBFS+ algorithm.

```
Procedure LBFS+(G, τ);
{ Input: a graph G and an ordering τ of V(G);
Output: an ordering σ of V(G)}
We run LBFS(G, τ(|V(G)|)). In the LBFS procedure at step (♣), let S be
the set of vertices with the lexicographically largest label. Now v is chosen
to be the vertex in S that appears last in τ.
```

Due to the many possible ways of breaking ties, namely choosing a vertex from a slice, the ordering returned by LBFS may not be unique. In contrast, the LBFS+(G,τ) algorithm will produce a unique

vertex ordering, with $\tau(|V(G)|)$ being the first vertex.

Let G be a graph and let $S \subseteq V(G)$. Throughout the paper, $G[S]$ denotes the subgraph of G induced by S and $G - S$ refers to the subgraph $G[V(G) \setminus S]$. We write $N_G(S)$ for $(\cup_{v \in S} N_G(v)) \setminus S$ and write $N_G[S]$ for $N_G(S) \cup S$. The set S is a *module* of G if $N_G(x) \setminus S = N_G(y) \setminus S$ for all $x, y \in S$. We say that u is a *good vertex* of G provided it is the last vertex in some LBFS ordering of G [8]. Let ρ be an LBFS ordering of G and let $S_\rho(k)$ be any slice of ρ. For any $u \in S_\rho(k)$, we say that u *flies with respect to* $S_\rho(k)$ *in* ρ provided $N_G(u) \setminus (\rho[k-1] \cup S_\rho(k)) \neq \emptyset$. In the following, we describe a new LBFS algorithm, which we call LBFS$^\square$.

Procedure LBFS$^\square$ (G);
{ Input: a graph G;
Output: an ordering ρ of $V(G)$}
Find a good vertex u of G and run LBFS(G, u). In the LBFS procedure at step (♣), let S be the set of vertices with the lexicographically largest label. Now v is chosen to be any vertex of S if $G[S]$ is a clique, v is chosen to be any good vertex of $G[S]$ if $G[S]$ is a module of G, else v is chosen to be any vertex of S which flies with respect to S in ρ.

Roughly speaking, the idea of LBFS$^\square$ is to orient a slice in the "right direction". Theorem 4.31 says that applying an LBFS$^\square$ sweep on an interval graph G, followed by an LBFS+, will produce an I-ordering. Since a good vertex of any graph G can be found by an LBFS, Theorem 1.3 says that this gives a 3-sweep LBFS recognition algorithm for interval graphs. Unfortunately, we do not see any direct linear time implementation of LBFS$^\square$.

To present our linear time LBFS recognition algorithm for interval graphs, we need to introduce a new variant of LBFS. Let G be a graph and σ an ordering of $V(G)$. For any $u \in V(G)$, we denote the number $\max\{j : \sigma(j) \in N_G[u]\}$ by $f_\sigma(u)$. It is obvious that $f_\sigma(u) \geq \sigma^{-1}(u)$.

Procedure LBFS$^\bigstar$(G, σ);
{ Input: a graph G and an ordering σ of $V(G)$;
Output: an ordering ρ of $V(G)$}
begin
  for each $u \in V(G)$ do $label(u) \leftarrow \emptyset$.
  for $i \leftarrow 1$ to $|V(G)|$ do begin
    if $i = 1$, $\rho(i) \leftarrow \sigma(|V(G)|)$;
    if $i > 1$, let S denote the set of vertices with the lexicographically largest label;
    let $\alpha = \min\{j : \sigma(j) \in S\}$ and $\beta = \max\{j : \sigma(j) \in S\}$.
    let w be any vertex of S such that $f_\sigma(w) = \max\{f_\sigma(x) : x \in S\}$.
      $\rho(i) \leftarrow \sigma(\alpha)$ when $L^{\alpha-1}_{G,\sigma}(\sigma(\alpha)) \setminus \rho[i-1] \neq \emptyset$;
      else $\rho(i) \leftarrow w$ when $f_\sigma(w) > \beta$ and $\rho(i) \leftarrow \sigma(\beta)$ when $f_\sigma(w) \leq \beta$;
    for each unvisited vertex u in $N_G(\rho(i))$ do
      $label(u) = label(u) \cup \{i\}$;
  end; {LBFS$^\bigstar$}

For any interval graph G, it worths mentioning that, if δ is an LBFS ordering of G, σ is an output of LBFS+(G, δ) and ρ an output of LBFS$^\bigstar(G)$, then ρ can be produced by LBFS$^\square$ (G), as shown in

Lemma 4.17. Though LBFS$^{\bigstar}$ looks more complicated than LBFS$^{\square}$, it can be implemented in linear time, as demonstrated in Section 5.

3 The algorithm

We are now in a position to spell out the details of our new algorithm for recognizing interval graphs. Once known Lemma 4.17 and Theorem 4.31, the following algorithm comes to our mind naturally.

Four-Sweep LBFS Recognition Algorithm(G):
{ Input: a graph G with n vertices;
Output: "G is a unit interval graph" and a UI-ordering of G if G is a unit interval graph; "G is an interval graph" and an I-ordering of G if G is an interval graph but not a unit interval graph; else, "G is not an interval graph".}
Step 1: Do an arbitrary LBFS on G to obtain an ordering δ;
Step 2: Do LBFS+(G, δ) yielding sweep σ. Calculate $L^{i-1}_{G,\sigma}(\sigma(i))$ and $f_\sigma(\sigma(i))$ for every $i \in [n]$;
Step 3: Do LBFS$^{\bigstar}$(G, σ) yielding sweep ρ;
Step 4: Do LBFS+(G, ρ) yielding sweep τ;
Step 5: If τ is a UI-ordering of G, return "G is a unit interval graph" and a UI-ordering τ of G;
Step 6: If τ is not a UI ordering but an I-ordering of G, return "G is an interval graph" and an I-ordering τ of G; else, return "G is not an interval graph".

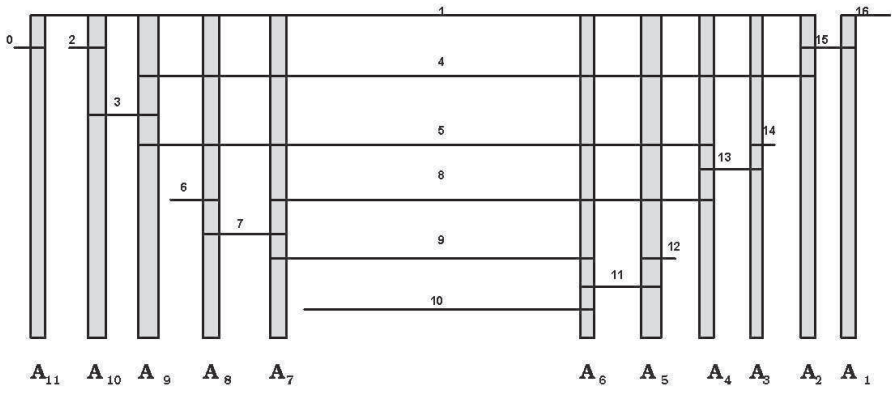

Fig. 1: The interval representation given by Tze-Heng Ma [39].

The main results of this paper, Theorems 4.30 and 4.32, state that the Four-Sweep LBFS Recognition Algorithm recognizes unit interval graphs and interval graphs correctly.

Example 3.1 *Here are four orderings obtained by the Four-Sweep* LBFS *Recognition Algorithm for the intersection graph of the seventeen intervals given in Fig. 1, in which we use brackets to indicate those nontrivial slices:*

δ: [0 1 [14 [4 13] [5 8] [7 9 [10 11] 12] 6 3 15 2] 16];
σ: [16 15 [4 1] [3 5 [6 7 [9 8] [11 10] 12 13] 14] 2 0];
ρ: [0 1 [15 4 [3 5 [13 8 [7 9 [11 10] 12] 6] 14] 2] 16];
τ: [16 15 [4 1] [14 13 [8 5] [12 11 [10 9] 7] 6 3] 2 0].

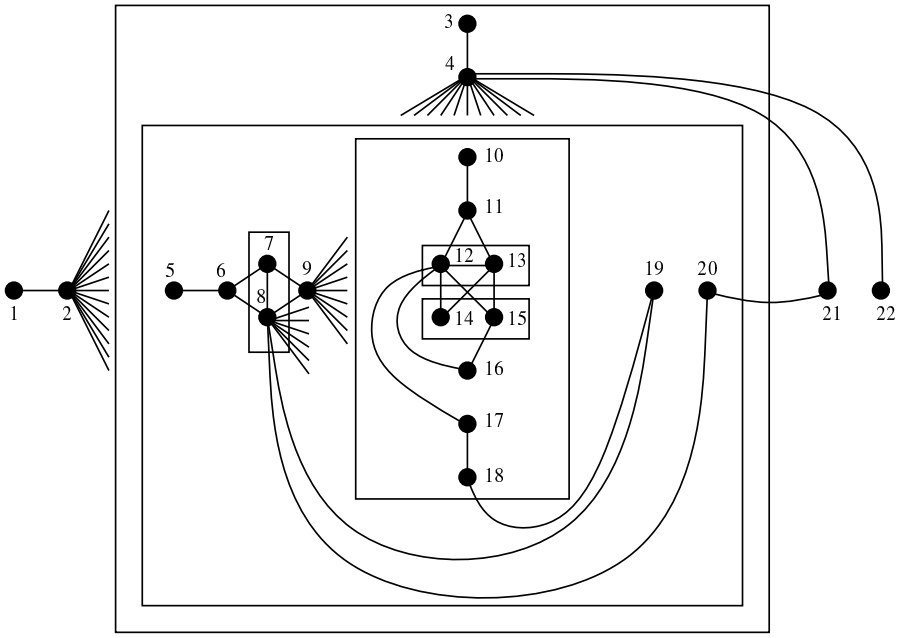

Fig. 2: The graph given by Corneil et al. [13, Figure 7].

Example 3.2 *We consider the interval graph G depicted in Fig. 2. Below are four orderings obtained by the Four-Sweep* LBFS *Recognition Algorithm on G:*

δ: [20 [8 [2 4] 21] [16 [15 [9 12]] 13 [11 14] 17 10 18 7 19 6] 5 3 1 22];
σ: [22 4 [3 2 [5 6 [7 8] 9 [18 17 12 [14 13 [11 15] 16] 10] 19 20] 21] 1];
ρ: [1 2 [4 [20 8 [6 7 9 [18 17 12 [11 13 [15 14] 16]] 19] 5 3]] 21 22];
τ: [22 4 [21 20 [8 2] [19 18 [17 9] 12 [16 15 13 [14 11]] 10 7 6] 5 3] 1].

Example 3.3 *We consider the same graph G as in Example 3.2. Here is another set of four orderings gleaned from the process of executing the Four-Sweep* LBFS *Recognition Algorithm on G:*

δ: [22 4 [3 2 [5 6 [7 8] 9 [18 17 12 [14 13 [11 15] 16] 10] 19 20] 21] 1];
σ: [1 2 [20 [8 4] [19 18 [17 9] 12 [16 15 13 [11 14]] 10 7 6] 5 3] 21 22];

ρ: [22 4 [2 [20 8 [6 7 9 [18 17 12 [11 13 [15 14] 16] 10] 19] 5 3] 21] 1];
τ: [1 2 [3 4 [5 6 [7 8] 9 [10 11 [13 12] [14 15] 16 17 18] 19 20]] 21 22].

A key underlying concept for our algorithm design and analysis is the *perfect orderings of interval graphs*. Let G be an interval graph with n vertices, let ψ be an ordering of $V(G)$ and let k be a nonnegative integer no greater than n. An interval representation $\mathcal{I}$ of G is *k-perfect with respect to ψ* provided

$$\ell_{\mathcal{I}}(\psi(1)) \leq \ell_{\mathcal{I}}(\psi(2)) \leq \cdots \leq \ell_{\mathcal{I}}(\psi(k)) \leq \min_{p\in[k+1,n]} \ell_{\mathcal{I}}(\psi(p)).$$

The ordering ψ is *k-perfect* if with respect to which G has a k-perfect interval representation. Note that ψ is always 0-perfect and it is n-perfect if and only if it is an I-ordering of G.

Recall that our four-sweep algorithm indeed generates four orderings, δ, σ, ρ and τ. Suppose the input graph is an interval graph on n vertices. The basic idea of our algorithm is to make all preparations step by step so that we can show for k from 1 to n inductively that the output ordering τ is guaranteed to be k-perfect.

4 Correctness of the algorithm

In order to establish the correctness of the algorithm, we have to go through a long line of lemmas.

4.1 General graphs

Lemma 4.1 *Let σ be an* LBFS *ordering of an n-vertex graph G. For any $i, j \in [n]$, if there is an element $k \in [n]$ such that $k < \min\{i,\ j\}$ and $L^k_{G,\sigma}(\sigma(j)) <_\sigma L^k_{G,\sigma}(\sigma(i))$, then $i < j$.*

Proof: Let $k' = \min\{i,\ j\}$. Since $k \leq k' - 1$, it follows from $L^k_{G,\sigma}(\sigma(j)) <_\sigma L^k_{G,\sigma}(\sigma(i))$ that $L^{k'-1}_{G,\sigma}(\sigma(j)) <_\sigma L^{k'-1}_{G,\sigma}(\sigma(i))$ and so the rule of LBFS gives $i < j$. □

Lemma 4.2 *Let G be a graph and let ψ be an* LBFS *ordering of G. Let S be a slice of ψ. Then there are integers i and k such that $\psi[i, k] = S$.*

Proof: It follows directly from Lemma 4.1. □

Lemma 4.3 *Let G be a graph and τ an* LBFS *ordering of G. Take $W \subseteq V(G)$ and suppose that C is a union of one or more connected components of $G - W$. Let $q = \min\{j : \tau(j) \in C\}$ and let p be a number such that $\tau(p) \notin W$ and $N_G(\tau(p)) \cap W \supseteq N_G(\tau(q)) \cap W$. Then $\tau[q, p] \subseteq S_\tau(q)$.*

Proof: If $q > p$, then $\tau[q, p] = \emptyset$ and so the claim is trivial. Now suppose $q \leq p$. The minimality of q implies that $L^{q-1}_{G,\tau}(\tau(q)) = N_G(\tau(q)) \cap W \cap \tau[q-1] \subseteq N_G(\tau(p)) \cap W \cap \tau[q-1] = L^{q-1}_{G,\tau}(\tau(p))$. In view of $q \leq p$ and the LBFS rule, we have $L^{q-1}_{G,\tau}(\tau(q)) = L^{q-1}_{G,\tau}(\tau(p))$. It is now immediate from Lemma 4.2 that $\tau[q, p] \subseteq S_\tau(q)$. □

Let G be a graph and let $W \subseteq V(G)$. We say that $v \in V(G)$ is *universal* with respect to W in G if v is adjacent to every vertex of $W \setminus \{v\}$. In general, we say $U \subseteq V(G)$ is *universal* with respect to W in G whenever every vertex from U is universal with respect to W in G.

Lemma 4.4 *Let G be a graph on n vertices and let τ be an* LBFS *ordering of G. Then the connected components of $G - N_G[\tau(n)]$, if any, can be listed as $C_1, \ldots, C_t$, in such a way that, for every $i \in [t]$, the vertices in $V(G) \setminus B_i$ appear in τ earlier than those in B_i, and $N_G(C_i)$ is universal with respect to B_i, where $B_i = (N_G[\tau(n)] \setminus N_G(C_i)) \cup (\cup_{j \in [i+1,t]} C_j)$. Moreover, for every $i \in [t]$, B_i is a module in G such that $N_G(B_i) = N_G(C_i)$.*

Proof: Let the number of connected components of $G - N_G[\tau(n)]$ be t. If G is a complete graph, then $t = 0$ and there is nothing to worry about. We hence assume that G is not a complete graph.

The rule of LBFS implies $V(G) \neq N_G[\tau(n)]$ and so $t \geq 1$ follows. We label the components of $G - N_G[\tau(n)]$ as $C_1, \ldots, C_t$ in a way so that

$$\alpha_1 < \cdots < \alpha_t \tag{1}$$

holds, where $\alpha_i = \min\{\tau^{-1}(x) : x \in C_i\}$.

Note first that $\tau(n) \in B_1$ and so the number $\min\{\tau^{-1}(v) : v \in B_1\}$ exists, which we denote by α. Applying Lemma 4.3 for $(W, C, q, p) = (N_G(C_1), B_1, \alpha, n)$ and $(N_G(C_1), C_1, \alpha_1, n)$, we get $\tau[\alpha, n] \subseteq S_\tau(\alpha)$ and $\tau[\alpha_1, n] \subseteq S_\tau(\alpha_1)$, respectively. This tells us

$$L_{G,\tau}^{\alpha-1}(v) = L_{G,\tau}^{\alpha-1}(\tau(n)) \tag{2}$$

for all $v \in \tau[\alpha, n]$ and

$$L_{G,\tau}^{\alpha_1-1}(\tau(\alpha_1)) = L_{G,\tau}^{\alpha_1-1}(\tau(n)). \tag{3}$$

An application of Eq. (3) leads to $\tau[\alpha_1 + 1, n] \supseteq N_G[\tau(n)] \setminus N_G(C_1)$, which, together with Eq. (1), shows $\alpha_1 < \alpha$. By the definition of C_1, $L_{G,\tau}^{\alpha-1}(\tau(n)) \cap C_1 = \emptyset$ and so Eq. (2) implies $L_{G,\tau}^{\alpha-1}(v) \cap C_1 = \emptyset$ for every $v \in \tau[\alpha, n]$. If $\tau[\alpha, n] \cap N_G[C_1] \neq \emptyset$, then, considering that $\alpha_1 < \alpha$ and that every vertex $u \in N_G[C_1]$ can be connected to $\tau(\alpha_1)$ by a path P so that $P - u$ lies in $G[C_1]$, we will be able to get a vertex $v \in \tau[\alpha, n] \cap N_G[C_1]$ satisfying $L_{G,\tau}^{\alpha-1}(v) \cap C_1 \neq \emptyset$, contradicting with what we just claim. This then gives $B_1 = \tau[\alpha, n]$. By virtue of Eq. (2) again now, we arrive at $N_G(C_1) = L_{G,\tau}^{\alpha-1}(\tau(n)) = L_{G,\tau}^{\alpha-1}(v) = N_G[v] \setminus B_1$ for all $v \in B_1$, proving the statement for $i = 1$.

Since B_1 is a module of G, the ordering τ^{B_1} of $G[B_1]$ induced by τ is an LBFS ordering. Hence, if $t > 1$, we can carry out the same proof for $i = 2$ as for $i = 1$ and in this way we can finish the proof for all $i \in [t]$ inductively. □

Let G be a graph. Two vertices u and v of G are *unrelated* with respect to $y \in V(G)$ if v and y lie in the same connected component of $G - N_G[u]$ and u and y lie in the same connected component of $G - N_G[v]$. The next lemma is a reformulation of [12, Theorem 4.1] and [8, Corollary 1.4].

Lemma 4.5 *Let G be a graph and let w be a good vertex of G. If u and v are two vertices unrelated with respect to w in G, then u, v and w form an asteroidal triple in G.*

Proof: Let C denote the connected component of $G - N_G[w]$ that contains u and D denote the connected component of $G - N_G[w]$ that contains v. The existence of C and D is guaranteed by the assumption that u and v are unrelated with respect to w. To prove the lemma, it suffices to show $C = D$.

Without loss of generality, assume that τ is an LBFS ordering of G such that $\tau^{-1}(u) < \tau^{-1}(v) < \tau^{-1}(w) = |V(G)|$. If $C \neq D$, then $N_G(C) \subseteq N_G(v)$ according to Lemma 4.4. Hence u and w cannot fall into the same connected component of $G - N_G[v]$, which is a contradiction to our assumption. □

Lemma 4.6 *Let G be a graph and let w be a good vertex of G. Let ρ be any* LBFS *ordering of G which starts at w and ends at a vertex u. Let v be a vertex such that w and u fall into the same connected component of $G - N_G[v]$. Then u, v and w form an asteroidal triple in G.*

Proof: By assumption, u and w fall into a connected component of $G-N_G[v]$. This gives $uv \notin E(G)$ and $wv \notin E(G)$. Moreover, applying Remark 2.1 yields $d_G(u,w) \geq d_G(v,w) \geq 2$. Accordingly, $G - N_G[u]$ has a connected component C which contains v and $G - N_G[u]$ has a connected component D which contains w.

Noting that $1 = \rho^{-1}(w) < \rho^{-1}(v) < \rho^{-1}(u) = |V(G)|$, we obtain from Lemma 4.4 that either $C = D$ or $N_G(D) \subseteq N_G(v)$. But the latter case is impossible, as we know that u and w fall into the same component of $G - N_G[v]$. So, we have $C = D$, proving that u and v are unrelated with respect to w in G. By now, the result follows from Lemma 4.5. □

Lemma 4.7 *Consider a graph G and an ordering σ of $V(G)$. Let ρ be an ordering obtained from* LBFS$^\bigstar(G, \sigma)$. *Suppose $S_\rho(\eta)$ is a slice of ρ and $\alpha = \min\{j : \sigma(j) \in S_\rho(\eta)\}$. If $L_{G,\sigma}^{\alpha-1}(\sigma(\alpha)) \setminus \rho[\eta-1] \neq \emptyset$, then $\rho(\eta)$ flies with respect to $S_\rho(\eta)$ in ρ.*

Proof: Since ρ is obtained from LBFS$^\bigstar(G, \sigma)$, it happens $\rho(\eta) = \sigma(\alpha)$. Owing to the minimality of α, it holds $L_{G,\sigma}^{\alpha-1}(\sigma(\alpha)) \cap S_\rho(\eta) = \emptyset$ and so $L_{G,\sigma}^{\alpha-1}(\sigma(\alpha)) \setminus (\rho[\eta-1] \cup S_\rho(\eta))$ is nothing but $L_{G,\sigma}^{\alpha-1}(\sigma(\alpha)) \setminus \rho[\eta-1] \neq \emptyset$. This proves that $\rho(\eta)$ flies with respect to $S_\rho(\eta)$ in ρ. □

Lemma 4.8 *Let ϕ and ψ be two* LBFS *orderings of a graph G. Take $s, p \in [|V(G)|]$. Assume that*

$$\psi(p) \in S_\phi(s) \tag{4}$$

and

$$N_G[x] \supseteq S_\phi(s) \tag{5}$$

for every $x \in S_\phi(s) \cap \psi[p-1]$. If

$$L_{G,\psi}^{p-1}(\psi(p)) \subseteq \phi[s-1] \cup S_\phi(s), \tag{6}$$

then $S_\phi(s) \setminus \psi[p-1] \subseteq S_\psi(p)$.

Proof: By Eq. (4), $N_G[x] \supseteq S_\phi(s)$ for every $x \in L_{G,\psi}^{p-1}(\psi(p)) \cap \phi[s-1]$; by Eq. (5), $N_G[x] \supseteq S_\phi(s)$ for every $x \in S_\phi(s) \cap \psi[p-1]$. Eq. (6) indeed gives us $L_{G,\psi}^{p-1}(\psi(p)) \subseteq \phi[s-1] \cup (S_\phi(s) \cap \psi[p-1])$. Henceforth, we know that $N_G[x] \supseteq S_\phi(s)$ for every $x \in L_{G,\psi}^{p-1}(\psi(p))$, which then implies $L_{G,\psi}^{p-1}(y) = L_{G,\psi}^{p-1}(\psi(p))$ for every $y \in S_\phi(s) \setminus \psi[p-1]$. This surely tells us $S_\phi(s) \setminus \psi[p-1] \subseteq S_\psi(p)$. □

4.2 Chordal graphs

Let G be a graph on n vertices. A vertex v of G is *simplicial* in G if $N_G[v]$ is a clique. An ordering ψ of $V(G)$ is a *perfect elimination ordering* (PEO) provided $\psi(k)$ is simplicial in the graph $G[\psi[k]]$ for every $k \in [n]$. We need an ordering characterization of chordal graphs.

Lemma 4.9 *[46] Let G be a graph and ψ be an* LBFS *ordering of G. Then G is chordal if and only if ψ is a PEO of G.*

Lemma 4.10 *Let G be a chordal graph and τ and ρ be two* LBFS *orderings of G. Suppose $S_\tau(k)$ is a non-clique slice of τ. Let $\eta = \min\{j : \rho(j) \in S_\tau(k), S_\tau(k) \setminus N_G[\rho(j)] \neq \emptyset\}$ and let $T = S_\tau(k) \setminus \rho[\eta - 1]$. If $\rho(\eta)$ does not fly with respect to $S_\tau(k)$ in τ, then*

$$T \subseteq S_\rho(\eta), \tag{7}$$

$S_\rho(\eta)$ is not a clique and $\rho(\eta)$ does not fly with respect to $S_\rho(\eta)$ in ρ.

Proof: Setting $s = k, p = \eta, \phi = \tau$ and $\psi = \rho$, the definition of η gives Eqs. (4) and (5) while the assumption that $\rho(\eta)$ does not fly with respect to $S_\tau(k)$ in τ gives

$$L_{G,\rho}^{\eta-1}(\rho(\eta)) \subseteq N_G(\rho(\eta)) \subseteq \tau[k-1] \cup S_\tau(k)$$

and hence Eq. (6) follows. Consequently, we can apply Lemma 4.8 to obtain Eq. (7). The fact that $S_\tau(k)$ is not a clique implies T is not a clique. Therefore, it follows from Eq. (7) that $S_\rho(\eta)$ is not a clique, as claimed.

We now turn to the last claim. By Lemma 4.2, $S_\rho(\eta) = \rho[\eta, \varsigma]$ for some $\varsigma \geq \eta$. Suppose to the contrary that $\rho(\eta)$ flies with respect to $S_\rho(\eta)$ in ρ, *i.e.*, it has a neighbor $\rho(t)$ for some $t > \varsigma$.

From Eq. (7) we have $\rho(t) \notin T$. This, together with the fact that $\rho(\eta)$ does not fly with respect to $S_\tau(k)$ in τ, shows that $\rho(t) \in \tau[k-1] \cup (S_\tau(k) \cap \rho[\eta - 1])$ and hence

$$T \subseteq N_G(\rho(t)). \tag{8}$$

Since we have assumed $t > \varsigma$, Eqs. (7) and (8) combine to say that $T \subseteq L_{G,\rho}^{t-1}(\rho(t))$, and then by Lemma 4.9, T is a clique, contradicting the known fact that T is not a clique. □

Lemma 4.11 *Let G be a chordal graph and let ψ be an* LBFS *ordering of G. Let S be a non-clique module and let $\beta = \max\{k : \psi(k) \in S\}$. Then $N_G(S) \subseteq \psi[\beta - 1] \setminus S$.*

Proof: Take $x \in N_G(S)$. As S is a module, $N_G(x) \supseteq S$. Moreover, note that S is not a clique and so we derive from Lemma 4.9 that $\psi^{-1}(x) < \beta$. □

Lemma 4.12 *Let G be a chordal graph and let σ be an* LBFS *ordering of G. Let ρ be the output of* LBFS$^\star(G, \sigma)$ *and let $S_\rho(\eta)$ be a slice of ρ. If $S_\rho(\eta)$ is a non-clique module of G, then $\rho(\eta) = \sigma(\beta)$ where $\beta = \max\{j : \sigma(j) \in S_\rho(\eta)\}$.*

Proof: Taking into account Lemma 4.2, an application of Lemma 4.11 on the non-clique module and the LBFS ordering of G, $S = S_\rho(\eta)$ and ρ, leads to

$$N_G(S) \subseteq \rho[\eta - 1]. \tag{9}$$

Making use of Lemma 4.11 on $S = S_\rho(\eta)$ and the LBFS ordering σ instead, we find

$$N_G(S) \subseteq \sigma[\beta - 1] \setminus S. \tag{10}$$

The rule of LBFS* says that there are three ways to determine the value of $\rho(\eta)$ according to different cases. Eq. (9) says that the first case does not happen and Eq. (10) says that the second case is also impossible. So, we fix $\rho(\eta)$ by the third way as described in LBFS* and that gives $\rho(\eta) = \sigma(\beta)$, as desired. □

Let G be a graph. We call $S \subseteq V(G)$ an *M-slice* (module of a slice) if S is a slice of G or S is a nontrivial module of the graph $G[T]$ for some slice T of G [13]. The following result, known as the Chordal LBFS Theorem, reveals some important aspects of the LBFS structure.

Lemma 4.13 *[13, Theorem 3.17] Let G be a chordal graph and let S be an M-slice of an arbitrary* LBFS *ordering of G. Then for any* LBFS *ordering τ of G, the ordering τ^S is an* LBFS *ordering of $G[S]$.*

Lemma 4.14 *Let ρ and δ be two* LBFS *orderings of a chordal graph G and let σ be the output of* LBFS$+(G,\delta)$. *Let $S_\rho(\eta)$ be a slice of ρ and suppose that $S_\rho(\eta)$ is not a clique. Let α and β be two integers such that*

$$\alpha = \min\{j : \sigma(j) \in S_\rho(\eta)\},\ \beta = \max\{j : \sigma(j) \in S_\rho(\eta)\}, \tag{11}$$

and

$$L^{\alpha-1}_{G,\sigma}(\sigma(\alpha)) \subseteq \rho[\eta - 1]. \tag{12}$$

Assume that there exists $v \in V(G)$ such that

$$v \in (N_G(S_\rho(\eta)) \cap \sigma[\beta]) \setminus \rho[\eta - 1]. \tag{13}$$

Then, v, $\sigma(\alpha)$ and $\sigma(\beta)$ form an asteroidal triple in G.

Proof: We set $p = \alpha, s = \eta, \phi = \rho$ and $\psi = \sigma$ in Lemma 4.8. By the definition of α in Eq. (11), we know that $\sigma(\alpha) \in S_\rho(\eta)$ and $S_\rho(\eta) \cap \sigma[\alpha - 1] = \emptyset$. The former establishes Eq. (4) while the latter says that Eq. (5) is vacuously true. Observe that Eq. (12) gives $L^{p-1}_{G,\psi}(\psi(p)) \subseteq \phi[s-1]$ and so Eq. (6) is guaranteed. Therefore, Lemma 4.8 ensures

$$S_\rho(\eta) = S_\rho(\eta) \setminus \sigma[\alpha - 1] \subseteq S_\sigma(\alpha). \tag{14}$$

In case that

$$\sigma^{-1}(v) \leq \alpha, \tag{15}$$

we can derive

$$\begin{array}{rcll} v & \in & N_G(S_\rho(\eta)) & \text{(By Eq. (13))} \\ & \subseteq & \sigma[\alpha-1] \cap N_G(S_\sigma(\alpha)) & \text{(By Eqs. (14) and (15))} \\ & = & L^{\alpha-1}_{G,\sigma}(\sigma(\alpha)) & \\ & \subseteq & \rho[\eta-1], & \text{(By Eq. (12))} \end{array}$$

contradicting Eq. (13). Noting from Eq. (13) that $\sigma^{-1}(v) < \beta$, we then find

$$\alpha < \sigma^{-1}(v) < \beta, \tag{16}$$

and henceforth, by Lemma 4.2,

$$v \in S_\sigma(\alpha). \tag{17}$$

By Eq. (11),

$$\{\sigma(\alpha), \sigma(\beta)\} \subseteq S_\rho(\eta). \tag{18}$$

Eq. (13) implies $v \notin S_\rho(\eta) \cup \rho[\eta - 1]$. Accordingly, as ρ is an LBFS ordering, there exists an integer q such that

$$\sigma(q) \in \rho[\eta - 1],$$

$$\{\sigma(q)\sigma(\alpha), \sigma(q)\sigma(\beta)\} \subseteq E(G) \tag{19}$$

and

$$\sigma(q)v \notin E(G). \tag{20}$$

Because $S_\rho(\eta)$ is a slice of G, Lemma 4.13 gives that $\sigma^{S_\rho(\eta)}$ is an LBFS ordering of $G[S_\rho(\eta)]$. Further recall that $S_\rho(\eta)$ is not any clique in G. We thus derive from Remark 2.1 and Eq. (11) that

$$d_G(\sigma(\alpha), \sigma(\beta)) > 1. \tag{21}$$

By Eqs. (17), (19) and (20), we have $\{v, \sigma(\alpha)\} \subseteq S_\sigma(\alpha)$ and $\sigma(q) \in N_G(\sigma(\alpha)) \setminus N_G(v)$ and so

$$q > \alpha \tag{22}$$

follows. Moreover, in light of Lemma 4.9, Eqs. (19), (21) and (22), we can obtain

$$q < \beta. \tag{23}$$

By Lemma 4.2, a direct consequence of Eqs. (14), (18), (22) and (23) is

$$\sigma(q) \in S_\sigma(\alpha). \tag{24}$$

Recall that σ is the output of LBFS+(G, δ). This combined with Lemmas 4.9 and 4.13 says that $\sigma(\alpha)$ is a simplicial vertex of $G[S_\sigma(\alpha)]$. It follows from Eqs. (17) and (24) that $\{v, \sigma(q)\} \subseteq S_\sigma(\alpha)$ and from Eqs. (19) and (20) that $\sigma(q) \in N_G(\sigma(\alpha)) \setminus N_G(v)$. This then shows that

$$v \notin N_G[\sigma(\alpha)]. \tag{25}$$

Lemma 4.9 asserts that $\sigma(\beta)$ is a simplicial vertex in $G[\sigma[\beta]]$. However, according to Eqs. (13) and (23), $\{v, \sigma(q)\} \subseteq \sigma[\beta]$; while as shown in Eqs. (19) and (20), $\sigma(\beta)\sigma(q) \in E(G)$ and $\sigma(q)v \notin E(G)$. The above consideration gives us

$$v \notin N_G[\sigma(\beta)]. \tag{26}$$

We are ready to verify that v, $\sigma(\alpha)$ and $\sigma(\beta)$ form an asteroidal triple in G. Observe that v, $\sigma(\alpha)$ and $\sigma(\beta)$ form an independent set of size three in G, as can be seen from Eqs. (21), (25) and (26).

By Eqs. (19), (20), (25) and (26), we see that $\sigma(\alpha), \sigma(q), \sigma(\beta)$ is a $\sigma(\alpha), \sigma(\beta)$-path inside $G - N_G[v]$.

We now want to show that $\sigma(\alpha)$ and v fall into the same connected component of $G - N_G[\sigma(\beta)]$. Assume the opposite. In view of Eq. (16) and Lemma 4.4, there are then two connected components C and C' of $H - N_H[\sigma(\beta)]$, where $H = G[\sigma[\beta]]$, such that $\sigma(\alpha) \in C$, $v \in C'$ and $N_H(C) \subseteq N_H(v)$. But Eqs. (19), (20) and (23) combine to say that $\sigma(q) \in N_H(C) \setminus N_H(v)$, a contradiction.

Finally, we need to show that $\sigma(\beta)$ and v fall into the same connected component of $G - N_G[\sigma(\alpha)]$. Observe that Eqs. (14), (17), (18) and (24) together implies

$$\sigma(\alpha), \sigma(\beta), v, \sigma(q) \in S_\sigma(\alpha). \tag{27}$$

Let $H = G[S_\sigma(\alpha)]$. To complete the proof, assume for contradiction that $\sigma(\beta)$ and v fall into different connected components of $G - N_G[\sigma(\alpha)]$, and so $H - N_H[\sigma(\alpha)]$ has two different connected components C and C' such that $v \in C$ and $\sigma(\beta) \in C'$. By Lemma 4.13, $\delta^{S_\sigma(\alpha)}$ is an LBFS ordering of H. As σ is the output of LBFS+(G, δ), it follows that $\sigma(\alpha)$ is a good vertex of H. We now conclude from Lemma 4.4 that either $N_H(v) \cap N_H(\sigma(\alpha)) \subseteq N_H(C) \subseteq N_H(\sigma(\beta))$ or $N_H(\sigma(\beta)) \cap N_H(\sigma(\alpha)) \subseteq N_H(C') \subseteq N_H(v)$. Since Eqs. (19) and (20) already guarantee $\sigma(q) \in (N_H(\sigma(\beta)) \cap N_H(\sigma(\alpha))) \setminus N_H(v)$, it must happen

$$N_H(v) \cap N_H(\sigma(\alpha)) \subsetneq N_H(\sigma(\beta)). \tag{28}$$

By Lemma 4.13, $\sigma^{S_\sigma(\alpha)}$ is an LBFS ordering of H starting from $\sigma(\alpha)$. Henceforth, as a consequence of Eqs. (25) and (28) as well as the LBFS rule, we get $\beta < \sigma^{-1}(v)$, violating Eq. (16), as desired. □

4.3 Interval graphs

We begin with a result which has a similar flavor to a key observation of Corneil *et al.* in developing their 6-sweep LBFS algorithm, the so-called Flipping Lemma [13, Lemma 4.4].

Lemma 4.15 *Let M be a module of an interval graph G. Let ρ be any* LBFS *ordering of G and τ be the output of* LBFS+(G, ρ). *Suppose that $\eta = \min\{j : \rho(j) \in M\}$. If $\rho(\eta)$ is a good vertex of $G[M]$, then $\tau^{-1}(\rho(\eta)) = \max\{j : \tau(j) \in M\}$.*

Proof: Since M is a module, ρ^M is an LBFS ordering of $G[M]$ and τ^M is the output of LBFS+$(G[M], \rho^M)$. Therefore, without loss of generality, we can assume that G is connected, $M = V(G)$ and $\eta = 1$. To obtain a contradiction, we suppose $\rho(1) = w$ and $\tau(n) = v \neq w$, where $n = |V(G)|$. Take $k \in [n]$ to be the largest possible number such that $\{v, w\} \subseteq S_\tau(k)$ and let $\tau(k) = u$.

Recall that $\rho(1) = w$ is assumed to be a good vertex of G and so, by Theorem 1.1 and Lemma 4.9, is simplicial in G. Therefore, the assumption of $vw \in E(G)$ and $n > \tau^{-1}(w)$ will ensure $L_{G,\tau}^{\tau^{-1}(w)-1}(v) = L_{G,\tau}^{\tau^{-1}(w)-1}(w)$. Considering that $w = \rho(1)$ and that τ is the output of LBFS+(G, ρ), we can continue to infer that w should be visited after $\tau(n) = v$. This is absurd and thus we get $vw \notin E(G)$ and hence w lies in some connected component of $G[S_\tau(k)] - N_{G[S_\tau(k)]}[v]$, say C.

In light of the rule of LBFS+(G, ρ), $\rho^{-1}(u) = \max\{j : \rho(j) \in S_\tau(k)\}$. Accordingly, as $S_\tau(k)$ is a slice of τ, Lemma 4.13 claims that $\rho(1) = w$ is a good vertex of $G[S_\tau(k)]$ while $\rho^{S_\tau(k)}$ is an LBFS ordering of $G[S_\tau(k)]$ going from w to u. By Theorem 1.1 and Lemma 4.6, we now have $\tau(k) = u \notin C$.

Let $s = \min\{j : \tau(j) \in C\}$. It is apparent that

$$s > k \tag{29}$$

and that

$$L_{G,\tau}^{s-1}(\tau(s)) \subseteq L_{G,\tau}^{s-1}(\tau(n)). \tag{30}$$

From Eq. (30) we derive $v = \tau(n) \in S_\tau(s)$. By invoking Lemma 4.2 additionally, we now deduce from $s \leq \tau^{-1}(w) < n$ that $\{v, w\} \subseteq S_\tau(s)$. This says that Eq. (29) contradicts with the maximality of k, as desired. □

Lemma 4.16 *Let G be an interval graph and let δ be an* LBFS *ordering of G. Let σ be the output of* LBFS+(G, δ) *and ρ an output of* LBFS$^\bigstar(G, \sigma)$. *If $S_\rho(\eta)$ is a non-clique slice of ρ such that $N_G(S_\rho(\eta)) \setminus \rho[\eta-1] \neq \emptyset$, then $\rho(\eta)$ flies with respect to $S_\rho(\eta)$ in ρ.*

Proof: Let $\alpha = \min\{j : \sigma(j) \in S_\rho(\eta)\}$ and $\beta = \max\{j : \sigma(j) \in S_\rho(\eta)\}$. Due to Lemma 4.7, we need only focus on the case that $L^{\alpha-1}_{G,\sigma}(\sigma(\alpha)) \subseteq \rho[\eta-1]$.

Since $S_\rho(\eta)$ is a slice of ρ with $N_G(S_\rho(\eta)) \setminus \rho[\eta-1] \neq \emptyset$, there exist u and v such that $u \in S_\rho(\eta)$ and $v \in N_G(u) \setminus (\rho[\eta-1] \cup S_\rho(\eta))$, that is, u flies with respect to $S_\rho(\eta)$ in ρ. Let $r = \max\{f_\sigma(x) : x \in S_\rho(\eta)\}$, which is surely no smaller than $\sigma^{-1}(v)$. By Theorem 1.1 and Lemma 4.14, we have $\sigma^{-1}(v) > \beta$, and hence

$$r > \beta. \tag{31}$$

By the rule of LBFS$^\bigstar$, $\rho(\eta) = w$ for some vertex

$$w \in S_\rho(\eta) \tag{32}$$

such that $f_\sigma(w) = r$. Note that our remaining task is to show that w flies with respect to $S_\rho(\eta)$ in ρ. Since

$$\sigma(r)w \in E(G), \tag{33}$$

we only need to show that $\sigma(r) \notin \rho[\eta-1] \cup S_\rho(\eta)$. It follows from Eq. (31) and the maximality assumption on β that $\sigma(r) \notin S_\rho(\eta)$. If $\sigma(r) \in \rho[\eta-1]$, Eqs. (32) and (33) take us to

$$N_G(\sigma(r)) \supseteq S_\rho(\eta). \tag{34}$$

But Lemma 4.9 asserts that σ is a PEO of G, which, along with Eqs. (31) and (34), shows that $S_\rho(\eta)$ is a clique, violating the assumption that $S_\rho(\eta)$ is not any clique, as desired. □

Lemma 4.17 *Let δ be an* LBFS *ordering of an interval graph G and let σ be the ordering produced by* LBFS+(G, δ). *Let ρ be the output of* LBFS$^\bigstar(G, \sigma)$. *Then ρ is an output of* LBFS$^\square$ *(G).*

Proof: Let $|V(G)| = n$. We need to verify for every $\eta \in [n]$ that the tie-breaking rule corresponding to LBFS$^\square$ can choose $\rho(\eta)$ to visit when LBFS$^\square$ has visited $\rho(1), \ldots, \rho(\eta-1)$ in that order.

If $S_\rho(\eta)$ is a clique, LBFS$^\square$ can visit any vertex among $S_\rho(\eta)$ and so we are done.

Suppose $S_\rho(\eta)$ is a non-clique module of G. By Lemma 4.12, it holds $\rho(\eta) = \sigma(\beta)$ where $\beta = \max\{j : \sigma(j) \in S_\rho(\eta)\}$. By Lemma 4.13, $\sigma^{S_\rho(\eta)}$ is an LBFS ordering of $G[S_\rho(\eta)]$ and so $\rho(\eta) = \sigma(\beta)$ is a good vertex of $G[S_\rho(\eta)]$. This means that $\rho(\eta)$ can be chosen from $S_\rho(\eta)$ by LBFS$^\square$.

Finally, assume that $S_\rho(\eta)$ is neither a clique nor a module of G. It follows from Lemma 4.16 that $\rho(\eta)$ flies with respect to $S_\rho(\eta)$ in ρ, completing the proof. □

4.4 Perfect orderings

The first few lemmas in this section aim to establish a characterization of "end vertices" in interval graphs, which amounts to be a characterization of 1-perfect orderings of interval graphs (Theorem 4.22). The subsequent work in this section tries to collect several sufficient conditions for a $(k-1)$-perfect ordering of an interval graph to be a k-perfect ordering.

Let $\mathcal{I}$ be an interval representation of a graph G and let $x, y \in V(G)$. Let us write $\mathcal{I}(x) < \mathcal{I}(y)$ if $r_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(y)$. For two subsets M and N of $V(G)$, we say that $M <_{\mathcal{I}} N$ if $\mathcal{I}(x) < \mathcal{I}(y)$ for each $x \in M$ and each $y \in N$. Let P be a v, w-path with vertex set T. We call P a *minimal* v, w-path if for any nonempty subset W of T, $G[T \backslash W]$ contains no v, w-path. It is straightforward that every shortest v, w-path is a minimal v, w-path.

Lemma 4.18 *Let $\mathcal{I}$ be an interval representation of a graph G. Let $x_0 = v, x_1, \ldots, x_k = w$ be a minimal v, w-path in G. If $\mathcal{I}(w) < \mathcal{I}(v)$, then we have $\ell_{\mathcal{I}}(x_i) < \ell_{\mathcal{I}}(v)$ and $r_{\mathcal{I}}(w) < r_{\mathcal{I}}(x_i)$ for every $i \in [k-1]$.*

Proof: Suppose that $\ell_{\mathcal{I}}(x_i) \geq \ell_{\mathcal{I}}(v)$ for some $i \in [k-1]$. Without loss of generality, we assume that i is the largest possible index such that $\ell_{\mathcal{I}}(v) \leq \ell_{\mathcal{I}}(x_i)$ holds. This gives $\ell_{\mathcal{I}}(x_{i+1}) < \ell_{\mathcal{I}}(v) \leq \ell_{\mathcal{I}}(x_i) \leq r_{\mathcal{I}}(x_{i+1})$ and hence $x_{i+1}v \in E(G)$. Therefore, we find a shorter v, w-path $v, x_{i+1}, x_{i+2}, \ldots, x_{k-1}, w$, yielding a desired contradiction.

By symmetry, we can also show that $r_{\mathcal{I}}(w) < r_{\mathcal{I}}(x_i)$ for every $i \in [k-1]$ and hence complete the proof. □

A vertex of a graph G is *admissible* provided no pair of vertices is unrelated with respect to it in G.

Lemma 4.19 *Let G be an interval graph and let $\mathcal{I}$ be an interval representation of G. Fix $v \in V(G)$. Take $A = \{w \in V(G) : \mathcal{I}(w) < \mathcal{I}(v)\}$ and $B = \{u \in V(G) : \mathcal{I}(v) < \mathcal{I}(u)\}$, which give a partition of $V(G) \setminus N_G[v]$. Suppose that $C = \{c_1, \ldots, c_k\}$ is a connected component of $G[A]$ such that $r_{\mathcal{I}}(c_1) \geq \cdots \geq r_{\mathcal{I}}(c_k)$ and $D = \{d_1, \ldots, d_m\}$ is a connected component of $G[B]$ such that $\ell_{\mathcal{I}}(d_1) \leq \cdots \leq \ell_{\mathcal{I}}(d_m)$. Then the following statements hold:*

(i) For every $w \in A$ with $d_G(w, v) \geq 3$ and every $u \in B$ with $d_G(u, v) \geq 3$, w and u are unrelated with respect to v.

(ii) If v is admissible, then either $N_G(D) \subseteq \cap_{i=1}^{k}(N_G(c_i) \cap N_G(v))$ or $N_G(C) \subseteq \cap_{i=1}^{m}(N_G(d_i) \cap N_G(v))$.

Proof: (i) By symmetry, it suffices to show that every shortest path P connecting w and v in G will be outside of $N_G[u]$. Suppose this is not true. Then there exists a vertex $x \in V(P) \cap N_G[u]$ such that $r_{\mathcal{I}}(x) \geq \ell_{\mathcal{I}}(u) > r_{\mathcal{I}}(v)$. But Lemma 4.18 says $\ell_{\mathcal{I}}(v) \geq \ell_{\mathcal{I}}(x)$. Thus, we find that v and x are adjacent and u, x, v is a u, v-path, getting to $d_G(u, v) \leq 2$, a contradiction.

(ii) We first observe the following:

$$\begin{cases} N_G(C) = N_G(c_1) \cap N_G(v) \supseteq \cdots \supseteq N_G(c_k) \cap N_G(v) = \cap_{i=1}^{k}(N_G(c_i) \cap N_G(v)); \\ N_G(D) = N_G(d_1) \cap N_G(v) \supseteq \cdots \supseteq N_G(d_m) \cap N_G(v) = \cap_{i=1}^{m}(N_G(d_i) \cap N_G(v)). \end{cases} \tag{35}$$

If the claim does not hold, Eq. (35) means that we can find $e \in (N_G(c_1) \cap N_G(v)) \backslash N_G[d_m]$ and $f \in (N_G(d_1) \cap N_G(v)) \backslash N_G[c_k]$. Because $G[C]$ is connected, there exists a path $h_1, h_2, \ldots, h_p$ in $G[C]$, where $h_1 = c_k$ and $h_p = c_1$. It is now easy to see that the path $h_1, h_2, \ldots, h_p, e, v$ connects c_k and v without visiting any vertex of $N_G[d_m]$. In the same manner, we can construct a d_m, v-path in $G - N_G[c_k]$ based on any d_m, d_1-path in $G[D]$ and the edges $d_1 f$ and fv. This shows that d_m and c_k are unrelated with respect to v in G, contradicting with the assumption that v is admissible in G. □

Lemma 4.20 *Let G be an interval graph and let v be an admissible vertex of G. Let $\mathcal{I}$, A, B and C be defined as in Lemma 4.19. Assume that $N_G(C) \cap N_G(v)$ is universal with respect to $N_G(v)$ in G. Then there exists an interval representation $\mathcal{I}'$ of G such that $\{x : \mathcal{I}'(x) < \mathcal{I}'(v)\} = A \setminus C$ and $\{x : \mathcal{I}'(x) > \mathcal{I}'(v)\} = B \cup C$.*

Proof: Without loss of generality, assume that $M_{\mathcal{I}} = \{\ell_{\mathcal{I}}(x), r_{\mathcal{I}}(x) : x \in V(G)\}$ consists of $2|V(G)|$ distinct points on the real line and $\min\{s - t : s > t,\ s, t \in M_{\mathcal{I}}\} \geq 3$.

Let $D_1, \ldots, D_q$ be the connected components of $G[B]$ where $D_1 <_{\mathcal{I}} \cdots <_{\mathcal{I}} D_q$. Denote $D_0 = \{v\}$ and $D_{q+1} = \emptyset$. It follows from Lemma 4.19 (ii) that there exists an integer $p \in [0, q]$ such that $\cap_{y \in D_i}(N_G(y) \cap N_G(v)) \supseteq N_G(C)$ for all $i \in [0, p]$ and $N_G(D_i) \subseteq \cap_{x \in C}(N_G(x) \cap N_G(v))$ for all $i \in [p + 1, q]$.

Since $N_G[D_p] \setminus N_G(C) \supseteq D_p \neq \emptyset$, we can define $a = \max\{r_{\mathcal{I}}(x) : x \in N_G[D_p] \setminus N_G(C)\} + 1$. We further let $c = \min\{\ell_{\mathcal{I}}(x) : x \in C\}$, $d = \max\{r_{\mathcal{I}}(x) : x \in C\}$ and

$$b = \begin{cases} \min\{\ell_{\mathcal{I}}(x) : x \in D_{p+1}\} - 1, & \text{if } p < q; \\ a + 1, & \text{if } p = q. \end{cases}$$

It is easy to see that $c < d < a < b$. We define a function f from the set $\mathbb{R}$ of reals to itself by putting

$$f(x) = \frac{(x - d)(b - a)}{c - d} + a$$

for every $x \in \mathbb{R}$. Note that f is an orientation-reversing affine transformation which maps

$$[c, d] = \cup_{x \in C} \mathcal{I}(x)$$

to

$$[f(d), f(c)] = [a, b].$$

We are now ready to define another interval representation $\mathcal{I}'$ of G as follows:

$$\mathcal{I}'(x) = \begin{cases} [f(r_{\mathcal{I}}(x)), f(\ell_{\mathcal{I}}(x)], & \text{if } x \in C; \\ [\ell_{\mathcal{I}}(x), f(\ell_{\mathcal{I}}(x))], & \text{if } x \in N_G(C) \setminus N_G(D_{p+1}); \\ \mathcal{I}(x), & \text{else.} \end{cases}$$

Utilizing the fact that $N_G(C)$ is universal with respect to $\cup_{i=0}^{p} N_G(D_i)$ in G, it is no hard to check that $\mathcal{I}'$ is still an interval representation of G. Moreover, it is clear that $\{x : \mathcal{I}'(x) < \mathcal{I}'(v)\} = A \setminus C$ and $\{x : \mathcal{I}'(v) < \mathcal{I}'(x)\} = B \cup C$, as wanted. □

Lemma 4.21 *Let G be an interval graph and let v be an admissible vertex of G. Let $\mathcal{I}$, A, B be defined as in Lemma 4.19. Then, at least one of $N_G(A) \cap N_G(v)$ and $N_G(B) \cap N_G(v)$ is universal with respect to $N_G(v)$ in G.*

Proof: We assume the opposite. Since $N_G(A) \cap N_G(v)$ is not universal with respect to $N_G(v)$ in G, there exist $x \in A$ and $\{a, b\} \subseteq N_G(v)$ such that $a \in N_G(x) \setminus N_G[b]$; since $N_G(B) \cap N_G(v)$ is not universal with respect to $N_G(v)$ in G, there exist $y \in B$ and $\{c, d\} \subseteq N_G(v)$ such that $c \in N_G(y) \setminus N_G[d]$.

Note that $r_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(v) < r_{\mathcal{I}}(a) < \ell_{\mathcal{I}}(b) < r_{\mathcal{I}}(v) < \ell_{\mathcal{I}}(y)$ and hence $x,\ a,\ v$ is an x, v-path avoiding $N_G[y]$. Similarly, $y,\ c,\ v$ is a y, v-path avoiding $N_G[x]$. These two paths cannot exist simultaneously as v is admissible in G, arriving at a contradiction. □

Let $\mathcal{I}$ be an interval representation of a graph G. The *conjugate* of $\mathcal{I}$, denoted $\overleftarrow{\mathcal{I}}$, is the interval representation of G such that $\overleftarrow{\mathcal{I}}(x) = [-r_{\mathcal{I}}(x), -\ell_{\mathcal{I}}(x)]$ for all $x \in V(G)$. It is easy to see that the first vertex of a 1-perfect ordering of an interval graph G must be admissible. The next theorem asserts that the converse is also true, which can be seen as a characterization of the "end vertices" in interval graphs in terms of interval representation. Note that Gimbel has provided a forbidden subgraph characterization of the end vertices of interval graphs [19, Theorem 1].

Theorem 4.22 *Let G be an interval graph and let $v \in V(G)$. Then v is an admissible vertex of G if and only if there exists an interval representation $\mathcal{I}'$ of G satisfying $\ell_{\mathcal{I}'}(v) = \min\{\ell_{\mathcal{I}'}(z) :\ z \in V(G)\}$, namely G has a 1-perfect ordering starting from v.*

Proof: We only prove the forward direction. Let $\mathcal{I}$, A, B and C be defined as in Lemma 4.19.

CASE 1: $A = \emptyset$ or $B = \emptyset$.

By considering $\overleftarrow{\mathcal{I}}$ if necessary, we may assume $A = \emptyset$. We can choose the required interval representation $\mathcal{I}'$ of G to be given by $\mathcal{I}'(u) = \mathcal{I}(u)$ for $u \in V(G) \setminus \{v\}$ and $\mathcal{I}'(v) = [\min_{u\in V(G)} \ell_{\mathcal{I}}(u), r_{\mathcal{I}}(v)]$.

CASE 2: We now assume that $A \neq \emptyset$ and $B \neq \emptyset$.

Lemma 4.21 shows that at least one of $N_G(A)\cap N_G(v)$ and $N_G(B)\cap N_G(v)$ is universal with respect to $N_G(v)$ in G. If necessary, by changing $\mathcal{I}$ to its conjugate $\overleftarrow{\mathcal{I}}$, we can always assume that $N_G(A) \cap N_G(v)$ is universal with respect to $N_G(v)$ in G.

Let $C_i,\ i \in [p]$, be the connected components of $G[A]$. By Lemma 4.20, there is an interval representation $\mathcal{I}_1$ of G such that $A_1 = \{x :\ \mathcal{I}_1(x) < \mathcal{I}_1(v)\} = A\backslash C_1$ and $B_1 = \{x :\ \mathcal{I}_1(x) > \mathcal{I}_1(v)\} = B \cup C_1$. We denote this interval representation $\mathcal{I}_1$ of G by $T(\mathcal{I}, C_1)$. In the same manner we can get an interval representation $\mathcal{I}_2$ of G such that $\mathcal{I}_2 = T(\mathcal{I}_1, C_2)$, and continuing this way we garner an interval representation $\mathcal{I}_p$ of G satisfying $A_p = \{x :\ \mathcal{I}_p(x) < \mathcal{I}_p(v)\} = \emptyset$. In view of Case 1, we conclude that the claim is true. □

Corollary 4.23 *Let G be an interval graph. Then every ordering of $V(G)$ starting from a good vertex of G is a 1-perfect ordering of G.*

Proof: By Lemma 4.5, every good vertex of an AT-free graph is admissible. So, the result comes from Theorems 1.1 and 4.22. □

Lemma 4.24 *Let G be an interval graph on n vertices and let τ be an ordering of $V(G)$. Take $k \in [n]$ and suppose that $\mathcal{I}$ is an interval representation of G which is $(k-1)$-perfect with respect to τ. Assume that $\tau[k, n]$ is the disjoint union of X_1, X_2 and D, where X_2 is a non-clique module of G, $X_1 \subseteq N_G(X_2)$ and $\ell_{\mathcal{I}}(x) > \ell_{\mathcal{I}}(y)$ holds for all $x \in D$ and $y \in X_2$. Let $X = X_1 \cup X_2$ and further assume that $\tau(k)$ is a good vertex of $G[X]$ and X is a module in $G[X \cup \tau[k-1]]$. Then τ is a k-perfect ordering of G.*

Proof: Since X_2 is not a clique, we have $a < b$ where $a = \min\{r_{\mathcal{I}}(z) :\ z \in X_2\}$ and $b = \max\{\ell_{\mathcal{I}}(z) :\ z \in X_2\}$.

Since $\tau(k)$ is a good vertex of $G[X]$, by Corollary 4.23, there is an interval representation $\mathcal{I}''$ of $G[X]$ such that

$$\ell_{\mathcal{I}''}(\tau(k)) = \min\{\ell_{\mathcal{I}''}(z) : \ z \in X\}. \tag{36}$$

Up to an orientation-preserving affine transformation, let us assume that

$$\mathcal{I}''(z) \subseteq [a, b] \tag{37}$$

for all $z \in X$. We now define a map $\mathcal{I}'$ on $V(G)$ specified by

$$\mathcal{I}'(z) = \begin{cases} \mathcal{I}(z), & \text{if } z \in \tau[k-1] \cup D; \\ \mathcal{I}''(z), & \text{if } z \in X_2; \\ [\ell_{\mathcal{I}''}(z), r_{\mathcal{I}}(z)], & \text{if } z \in X_1. \end{cases} \tag{38}$$

CLAIM: The map $\mathcal{I}'$ is still an interval representation of G, namely

$$\mathcal{I}'(x) \cap \mathcal{I}'(y) \neq \emptyset \ \text{ if and only if } \ \mathcal{I}(x) \cap \mathcal{I}(y) \neq \emptyset \tag{39}$$

for all $x, y \in V(G)$. Note that $\mathcal{I}(x) \cap \mathcal{I}(y) \neq \emptyset$ if and only if either $xy \in E(G)$ or $x = y$.

We divide the proof of the Claim into six possible cases, depending on the choice of (x, y).

CASE 1: $(x, y) \in ((\tau[k-1] \cup D) \times (\tau[k-1] \cup D)) \cup (X_2 \times X_2)$.

This is trivial.

CASE 2: $(x, y) \in X_1 \times X_1$.

As X_2 is a module of G, we know that

$$xy \in E(G) \tag{40}$$

for all $x \in N_G(X_2)$ and $y \in X_2$ and hence especially

$$[a, b] \subseteq \mathcal{I}(x) \tag{41}$$

for all $x \in N_G(X_2)$. Recalling $X_1 \subseteq N_G(X_2)$, Eq. (41) implies

$$r_{\mathcal{I}'}(x) = r_{\mathcal{I}}(x) \geq b \tag{42}$$

for $x \in X_1$. But Eqs. (37) and (38) tell us

$$\ell_{\mathcal{I}'}(x) = \ell_{\mathcal{I}''}(x) \leq b \tag{43}$$

for every $x \in X_1$. We thus see that

$$b \in \mathcal{I}'(x) \subseteq \mathcal{I}(x) \tag{44}$$

for every $x \in X_1$ and hence Eq. (39) holds for all $(x, y) \in X_1 \times X_1$.

CASE 3: $(x, y) \in (X_1 \times X_2) \cup (X_2 \times X_1)$.

By symmetry, we only consider the case that $x \in X_1$ and $y \in X_2$. By virtue of $X_1 \subseteq N_G(X_2)$ again, Eq. (40) asserts that $xy \in E(G)$. On the other hand, we read from Eq. (38) that $\mathcal{I}'(y) = \mathcal{I}''(y)$ and we find from Eqs. (37), (43) and (44) that $\mathcal{I}'(x) \supseteq \mathcal{I}''(x)$. Consequently, as $xy \in E(G)$ and $\mathcal{I}''$ is an interval representation of $G[X]$, $\mathcal{I}'(x) \cap \mathcal{I}'(y) \supseteq \mathcal{I}''(x) \cap \mathcal{I}''(y) \neq \emptyset$. This then confirms the Claim in this case.

CASE 4: $(x,y)\in(X_2\times D)\cup(D\times X_2)$.

Without loss of generality, we assume $x\in D$ and $y\in X_2$. Our assumption on D gives

$$b=\max\{\ell_{\mathcal{I}}(z):\ z\in X_2\}<\ell_{\mathcal{I}}(x). \tag{45}$$

Comparing this with Eq. (41) leads to $xy\notin E(G)$. Meanwhile, Eqs. (37) and (38) imply $\mathcal{I}'(y)\subseteq[a,b]$ but Eqs. (38) and (45) give

$$b<\ell_{\mathcal{I}}(x)=\ell_{\mathcal{I}'}(x). \tag{46}$$

It follows $\mathcal{I}'(x)\cap\mathcal{I}'(y)=\emptyset$, as was to be shown.

CASE 5: $(x,y)\in(X_1\times D)\cup(D\times X_1)$.

Let us assume that $x\in D$ and $y\in X_1$. By Eq. (44), $b\in\mathcal{I}'(y)\subseteq\mathcal{I}(y)$. This along with Eqs. (38) and (46) means that $\mathcal{I}(x)\cap\mathcal{I}(y)\neq\emptyset$ and $\mathcal{I}'(x)\cap\mathcal{I}'(y)\neq\emptyset$ are both equivalent to the condition $r_{\mathcal{I}}(y)\geq\ell_{\mathcal{I}}(x)$, thereby verifying the Claim.

CASE 6: $(x,y)\in(X\times\tau[k-1])\cup(\tau[k-1]\times X)$.

We may assume that $x\in\tau[k-1]$ and $y\in X$. Since $\mathcal{I}$ is $(k-1)$-perfect with respect to τ, we know that

$$\ell_{\mathcal{I}}(x)\leq a. \tag{47}$$

First suppose $xy\notin E(G)$. It must hold $x\notin N_G[X]$ as X is a module in $G[X\cup\tau[k-1]]$. This implies that $a\notin\mathcal{I}(x)$ and hence, in view of Eqs. (38) and (47), $r_{\mathcal{I}'}(x)=r_{\mathcal{I}}(x)<a$. By Eqs. (37) and (38), $\ell_{\mathcal{I}'}(y)=\ell_{\mathcal{I}''}(y)\geq a$. Henceforth, $\mathcal{I}'(x)\cap\mathcal{I}'(y)=\emptyset$, as expected.

Next consider the case that $xy\in E(G)$. Applying the assumption that X is a module in $G[X\cup\tau[k-1]]$, we conclude that $N_G(x)\supseteq X$ and so $\mathcal{I}'(x)=\mathcal{I}(x)\supseteq[a,b]$. It is clear now that Eqs. (37) and (38) demonstrate that $\mathcal{I}'(x)\cap\mathcal{I}'(y)\supseteq[a,b]\cap\mathcal{I}'(y)\neq\emptyset$. This then completes our proof of the Claim.

From the Claim it is but a short step to the asserted result in the lemma. Indeed, for any $x\in D$, Eqs. (37), (38) and (45) tell us

$$\ell_{\mathcal{I}'}(\tau(k))=\ell_{\mathcal{I}''}(\tau(k))\leq b<\ell_{\mathcal{I}}(x)=\ell_{\mathcal{I}'}(x).$$

Consequently, from Eqs. (36), (38), and the assumption that $\mathcal{I}$ is $(k-1)$-perfect with respect to τ, we can conclude that $\mathcal{I}'$ is k-perfect with respect to τ. This means that τ is a k-perfect ordering of G and hence we are done. □

Lemma 4.25 *Let G be a graph on n vertices and pick $k\in[n]$. Assume that $\mathcal{I}$ is a $(k-1)$-perfect interval representation of G with respect to an* LBFS *ordering τ of $V(G)$. Then for each $x\in S_\tau(k)$ and each $y\in V(G)\setminus(S_\tau(k)\cup\tau[k-1])$, it holds $\ell_{\mathcal{I}}(x)<\ell_{\mathcal{I}}(y)$, $\tau^{-1}(x)<\tau^{-1}(y)$ and $N_G(y)\cap\tau[k-1]\subsetneq N_G(x)\cap\tau[k-1]=N_G(S_\tau(k))\cap\tau[k-1]$.* □

Lemma 4.26 *Let G be an interval graph on n vertices and take $k\in[n]$. Suppose that an interval representation $\mathcal{I}$ of G is $(k-1)$-perfect with respect to an ordering τ of $V(G)$. Let X be a subset of $V(G)$ such that $\tau(k)\in X\subseteq\tau[k,n]$. Let $D=\tau[k,n]\setminus X$. Assume that $\ell_{\mathcal{I}}(x)<\ell_{\mathcal{I}}(y)$ for every $x\in X$ and $y\in D$. If $G[X]$ is a complete graph and X is a module in $G[X\cup\tau[k-1]]$, then τ is k-perfect.*

Proof: Assume that $c=\min\{\ell_{\mathcal{I}}(x):\ x\in X\}$. We define an interval representation $\mathcal{I}'$ of G as follows: $\mathcal{I}'(x)=[c,r_{\mathcal{I}}(x)]$ for $x\in X$ and $\mathcal{I}'(x)=\mathcal{I}(x)$ else. It can be checked that $\mathcal{I}'$ is k-perfect with respect to τ and so τ is a k-perfect ordering of G. □

Lemma 4.27 *Let G be an interval graph on n vertices. Take $k \in [n]$, let τ be a $(k-1)$-perfect* LBFS *ordering of G, and pick $T \subseteq S_\tau(k)$. Suppose that $\tau(k)$ is a good vertex of $G[S_\tau(k)]$ and that $S_\tau(k) \setminus T$ is universal with respect to $S_\tau(k)$ in G. Then τ is a k-perfect ordering of G provided $S_\tau(k)$ is a clique or T is a module of G.*

Proof: Let $X = S_\tau(k)$, $D = \tau[k,n] \setminus X = V(G) \setminus (S_\tau(k) \cup \tau[k-1])$ and $\mathcal{I}$ be a $(k-1)$-perfect interval representation of G with respect to τ. By Lemma 4.25, $\ell_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(y)$ for every $x \in X$ and every $y \in D$. If X is a clique in G, then Lemma 4.26 implies that τ is k-perfect. So assume that X is not a clique, which means that T is a non-clique module of G. Setting $X_1 = S_\tau(k) \setminus T$ and $X_2 = T$, an application of Lemma 4.24 then finishes the proof. □

Lemma 4.28 *Let G be a graph on n vertices. Take $k \in [n]$. Let τ be a $(k-1)$-perfect* LBFS *ordering of G, let $m = |S_\tau(k)|$ and let $\mathcal{I}$ be a $(k-1)$-perfect interval representation of G with respect to τ. Suppose there exists $u \in S_\tau(k)$ such that $\mathcal{I}(\tau(k)) < \mathcal{I}(u)$ and there is an* LBFS *ordering σ of $G[S_\tau(k)]$ such that $\sigma(1) = u$ and $\sigma(m) = \tau(k)$. Then τ is a k-perfect ordering of G.*

Proof: Denote $\tau(k)$ by w and denote $G[S_\tau(k)]$ by H. Let $A = \{v : \mathcal{I}(v) < \mathcal{I}(w), v \in S_\tau(k)\}$, $B = \{v : \mathcal{I}(v) > \mathcal{I}(w), v \in S_\tau(k)\}$ and the connected components of $G[B]$ be $\{D_1, \ldots, D_t\}$ with $D_1 <_{\mathcal{I}} \cdots <_{\mathcal{I}} D_t$. Because of $\mathcal{I}(\tau(k)) < \mathcal{I}(u)$, we can assume that $u \in D_j$ for some $j \in [t]$. Let $D_0 = \{w\}$,

$$X = A \cup (\cup_{i \in [0,j-1]} N_H[D_i]) \setminus N_H(D_j)$$

and

$$D = N_H(D_j) \cup (\cup_{i \in [j,t]} D_i) \cup \tau[k+m,n]. \tag{48}$$

Observe that

$$N_H(D_i) \subseteq N_H(D_j) \tag{49}$$

for every $i \in [j,t]$. It is also easy to see that

$$V(G) = \tau[k-1] \cup X \cup D. \tag{50}$$

If X is a clique in G, the result is immediate from Lemma 4.26. So, let us assume that X is a not a clique. If we can show that X is a module of G, then Lemma 4.13 demonstrates that $\tau(k)$ is a good vertex of $G[X]$, and so, by putting $X_1 = \emptyset$ and $X_2 = X$, we can apply Lemma 4.24 to complete the proof.

It remains to show that X is a module of G. Let a and b be two vertices of X. On account of Eqs. (48) and (50), our task is to check the following:

$$\begin{cases} N_G(a) \cap \tau[k-1] = N_G(b) \cap \tau[k-1]; \\ N_G(a) \cap N_H(D_j) = N_G(b) \cap N_H(D_j); \\ N_G(a) \cap (\cup_{i \in [j,t]} D_i) = N_G(b) \cap (\cup_{i \in [j,t]} D_i); \\ N_G(a) \cap \tau[k+m,n] = N_G(b) \cap \tau[k+m,n]. \end{cases}$$

We do this one by one as follows.

- As $a, b \in S_\tau(k)$, $N_G(a) \cap \tau[k-1] = N_G(b) \cap \tau[k-1]$.

- Note that $a, b \notin N_H[D_j]$. Thus, it follows from Eq. (49) that $N_G(a) \cap D_i = N_G(b) \cap D_i = \emptyset$ for every $i \in [j, t]$.
- Employing Lemma 4.4 for the graph $G[S_\tau(k)]$, we have $N_H(D_j) \subseteq N_H(x)$ for all $x \in S_\tau(k) \setminus N_H[D_j]$. So, as $X \subseteq S_\tau(k) \setminus N_H[D_j]$, it holds $N_G(a) \cap N_H(D_j) = N_G(b) \cap N_H(D_j) = N_H(D_j)$.
- Since $u \in S_\tau(k)$, it is direct from Lemma 4.25 that $\ell_{\mathcal{I}}(u) < \ell_{\mathcal{I}}(y)$ for each $y \in \tau[k+m, n]$. But for every $x \in X$, we also have $r_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(u)$ and hence $N_G(x) \cap \tau[k+m, n] = \emptyset$. This shows $N_G(a) \cap \tau[k+m, n] = N_G(b) \cap \tau[k+m, n] = \emptyset$. □

Lemma 4.29 *Let G be an interval graph on n vertices, let ρ be an* LBFS *ordering of G and let τ be the output of* LBFS+(G, ρ). *Take $k \in [2, n]$. Assume that $S_\tau(k)$ is not a clique and τ is a $(k-1)$-perfect ordering of G. Let*

$$\eta = \min\{j : \rho(j) \in S_\tau(k), S_\tau(k) \setminus N_G[\rho(j)] \neq \emptyset\}. \tag{51}$$

If $\rho(\eta)$ flies with respect to $S_\tau(k)$ in τ, then τ is a k-perfect ordering of G.

Proof: Let $\beta = \max\{j : \rho(j) \in S_\tau(k)\} = \rho^{-1}(\tau(k))$, $D = V(G) \setminus (S_\tau(k) \cup \tau[k-1])$ and $\mathcal{I}$ be a $(k-1)$-perfect interval representation of G with respect to τ.

CLAIM 1: For every $x \in S_\tau(k) \setminus N_G[\rho(\eta)]$, it holds

$$N_G[x] \subseteq S_\tau(k) \cup \tau[k-1]. \tag{52}$$

By the assumption that $\rho(\eta)$ flies with respect to $S_\tau(k)$ in τ, we can find $y \in D \cap N_G(\rho(\eta))$. It follows from Lemma 4.25 that $\ell_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(y)$ and it follows from $y\rho(\eta) \in E(G)$ that $\ell_{\mathcal{I}}(y) \leq r_{\mathcal{I}}(\rho(\eta))$. Therefore, we have $\ell_{\mathcal{I}}(x) < r_{\mathcal{I}}(\rho(\eta))$. In view of $x \notin N_G[\rho(\eta)]$, this implies $\mathcal{I}(x) < \mathcal{I}(\rho(\eta))$, which further leads to

$$r_{\mathcal{I}}(x) < \ell_{\mathcal{I}}(\rho(\eta)). \tag{53}$$

Applying Lemma 4.25 again, we know that

$$\ell_{\mathcal{I}}(\rho(\eta)) < \ell_{\mathcal{I}}(z) \tag{54}$$

for all $z \in D$. Combining Eqs. (53) and (54) yields $xz \notin E(G)$ for all $z \in D$ and so Eq. (52) follows.

CLAIM 2: $\mathcal{I}(\rho(\beta)) < \mathcal{I}(\rho(\eta))$.

Owing to Eq. (51), there exists

$$x \in S_\tau(k) \setminus N_G[\rho(\eta)]. \tag{55}$$

Note that $\eta < \rho^{-1}(x) \leq \beta$ and so Lemma 4.1 gives

$$L^{\eta}_{G,\rho}(x) \geq_\rho L^{\eta}_{G,\rho}(\rho(\beta)) \tag{56}$$

and

$$L^{\eta-1}_{G,\rho}(x) \geq_\rho L^{\eta-1}_{G,\rho}(\rho(\beta)). \tag{57}$$

By Claim 1, Eq. (52) holds and hence, by the minimality of η and that $S_\tau(k)$ is a slice of τ,

$$L^{\eta-1}_{G,\rho}(x) = (\rho[\eta-1] \cap S_\tau(k)) \cup (\rho[\eta-1] \cap L^{k-1}_{G,\tau}(x)) \subseteq L^{\eta-1}_{G,\rho}(y)$$

for each $y \in S_\tau(k)$. Compared with Eq. (57), we arrive at

$$L^{\eta-1}_{G,\rho}(x) = L^{\eta-1}_{G,\rho}(\rho(\beta)). \tag{58}$$

This enables us to obtain

$$\rho(\eta)\rho(\beta) \notin E(G), \tag{59}$$

as otherwise we will deduce from Eqs. (55) and (58) that $L^{\eta}_{G,\rho}(x) \subsetneq L^{\eta}_{G,\rho}(\rho(\beta))$, a contradiction with Eq. (56). Owing to Eq. (59), we have either $\mathcal{I}(\rho(\beta)) < \mathcal{I}(\rho(\eta))$ or $\mathcal{I}(\rho(\eta)) < \mathcal{I}(\rho(\beta))$. Our remaining task is to exclude the latter possibility. Suppose it happens $\mathcal{I}(\rho(\eta)) < \mathcal{I}(\rho(\beta))$. As $\rho(\eta)$ flies with respect to $S_\tau(k)$ in τ, we find that there is a $z \in D$ such that $\ell_{\mathcal{I}}(z) \leq r_{\mathcal{I}}(\rho(\eta)) < \ell_{\mathcal{I}}(\rho(\beta))$, violating Lemma 4.25. This proves Claim 2.

By Lemma 4.13, the ordering $\rho_1 = \rho^{S_\tau(k)}$ induced by ρ is an LBFS ordering of $G[S_\tau(k)]$. We now define a new ordering σ of $G[S_\tau(k)]$ by requiring $\sigma(1) = \rho(\eta)$, $\sigma^{-1}(x) < \sigma^{-1}(y)$ for any $x \in S_\tau(k) \cap \rho[\eta-1]$ and any $y \in S_\tau(k) \setminus \rho[\eta]$, and $\sigma^{-1}(y_1) < \sigma^{-1}(y_2)$ if and only if $\rho_1^{-1}(y_1) < \rho_1^{-1}(y_2)$ for any $y_1, y_2 \in S_\tau(k) \setminus \rho[\eta]$. On account of Eq. (51), σ is an LBFS ordering of $G[S_\tau(k)]$ which starts at $\rho(\eta)$ and ends at $\rho(\beta) = \tau(k)$. Because of Claim 2, Lemma 4.28 applies to say that τ is a k-perfect ordering of G, as was to be shown. □

4.5 Main results

Our main algorithm is designed to recognize both unit interval graphs and interval graphs. Correspondingly, we need to verify the correctness of the algorithm with regard to these two recognition tasks.

An MNS type algorithm is an algorithm which visits the vertices of a graph by always choosing one whose set of neighbors already explored is maximal with respect to set inclusion [9]. An ordering produced by an MNS type algorithm is called an MNS ordering. It is clear that generic LBFS and many of its variants by introducing some additional tie-breaking rules are all MNS type algorithms and all LBFS orderings are MNS orderings. In [35], among other things we propose a 3-sweep MNS algorithm for recognizing connected unit interval graphs; see [35, Theorem 9]. We refer the reader to [35] for more information about the 3-sweep MNS algorithm presented there.

Theorem 4.30 *The Four-Sweep* LBFS *Recognition Algorithm recognizes unit interval graphs correctly.*

Proof: We follow the notation used in the algorithm description. Let G be a unit interval graph. By Theorem 1.4, we need to prove that the output ordering τ of the Four-Sweep LBFS Recognition Algorithm(G) is a UI-ordering of G. Note that σ is an MNS ordering of G, ρ is an MNS ordering of G starting from $\sigma(n)$ and τ is the output of MNS+(G, ρ). In the case that G is connected, we deduce from [35, Theorem 9] that τ is a UI-ordering of G. If G has more than one connected components, checking the rule of LBFS* shows that the last three sweeps of LBFS in our algorithm visit the connected components one by one and so the final ordering τ is still a UI-ordering of G. □

Theorem 4.31 *Let ρ be an output of* LBFS$^\square$ *(G) where G is an interval graph with n vertices. Let τ be the ordering produced by* LBFS+(G, ρ). *Then τ is an I-ordering of G.*

Proof: We intend to show that τ is n-perfect. Indeed, we shall proceed by induction on k to show that τ is a k-perfect ordering of G for each $k \in [n]$. Corollary 4.23 confirms the result for $k = 1$. Suppose

then $n \geq k > 1$ and that τ is $(k-1)$-perfect. It follows from Lemma 4.13 that $\tau(k)$ is a good vertex of $G[S_\tau(k)]$. Let

$$\eta = \max\{j : \rho(j) \in S_\tau(k), \cap_{x \in S_\tau(k) \cap \rho[j-1]} N_G[x] \supseteq S_\tau(k)\} \tag{60}$$

and let $T = S_\tau(k) \setminus \rho[\eta - 1]$.

If $S_\tau(k)$ is a clique of G or T is a module of G, then we infer from Lemma 4.27 that τ is a k-perfect ordering of G.

We now assume that $S_\tau(k)$ is not a clique and T is not a module of G. Note that in this case Eqs. (51) and (60) give the same definition for η. Since T is not a module, there exists

$$z \in N_G(T) \setminus (\tau[k-1] \cup S_\tau(k)) \tag{61}$$

such that

$$T \setminus N_G(z) \neq \emptyset. \tag{62}$$

If $\rho(\eta)$ flies with respect to $S_\tau(k)$ in τ, we may invoke Lemma 4.29 to establish that τ is a k-perfect ordering of G.

It remains to show that $\rho(\eta)$ does fly with respect to $S_\tau(k)$ in τ. If this were not true, Lemma 4.10 gives

$$T \subseteq S_\rho(\eta), \tag{63}$$

$S_\rho(\eta)$ is not a clique and $\rho(\eta)$ does not fly with respect to $S_\rho(\eta)$ in ρ. By the rule of $\texttt{LBFS}^\square$ (G),

$$S_\rho(\eta) \text{ is a module of } G \text{ and } \rho(\eta) \text{ is a good vertex of } S_\rho(\eta). \tag{*}$$

The claim (*) enables us obtain from Lemma 4.15 that

$$\tau^{-1}(\rho(\eta)) = \max\{j : \tau(j) \in S_\rho(\eta)\}. \tag{64}$$

It follows from Eq. (60) that $\rho(\eta) \in S_\tau(k)$. Thus, Eq. (64) combined with Lemma 4.2 gives

$$S_\rho(\eta) \subseteq \tau[k-1] \cup S_\tau(k). \tag{65}$$

Comparing Eqs. (61) and (65) leads to $z \in N_G(T) \setminus S_\rho(\eta)$ and hence

$$\begin{array}{rcll} N_G(z) & \supseteq & S_\rho(\eta) & \text{(By Eq. (63) and the first part of claim (*))} \\ & \supseteq & T, & \text{(By Eq. (63))} \end{array}$$

violating Eq. (62). This ends the proof. □

Theorem 4.32 *The Four-Sweep* LBFS *Recognition Algorithm recognizes interval graphs correctly.*

Proof: Let G be an interval graph with n vertices. Let δ, σ, ρ and τ be as described in the Four-Sweep LBFS Recognition Algorithm(G). By Lemma 4.17 and Theorem 4.31, τ is an I-ordering of G and so Theorem 1.3 guarantees that G is an interval graph. □

5 Linear time implementation

To close the paper, let us show that the Four-Sweep LBFS Recognition Algorithm can be implemented in linear time. We note that the generic LBFS and LBFS+ can both be implemented in linear time [5, 13, 20, 21, 46]. In particular, at each step of making a transversal of a graph G, say after visiting a subset $T \subseteq V(G)$, we can make use of partition refinement [21] to determine the LBFS slice S in $V(G) \setminus T$ and choose the vertex of S which appears last in the ordering ϕ (or appears earliest in the reversal of ϕ) in $O(1)$ time. That is, we can pick a vertex in an LBFS slice with respect to a given ordering ϕ in $O(1)$ time. Also observe that it is easy to check in linear time whether or not an ordering is a UI-ordering or/and an I-ordering [4, 13].

Let the input graph G have n vertices and m edges. For each $x \in V(G)$, let $\deg_G(x)$ stand for the size of $N_G(x)$. Our remaining task is to give a linear time implementation of LBFS$^\bigstar(G, \sigma)$ for producing the ordering ρ.

After we get the ordering σ, we calculate the numbers $\deg_G(u)$ and $f_\sigma(u) = \max\{j : \sigma(j) \in N_G[u]\}$ for all $u \in V(G)$. This can be done in $O(\sum_{u \in V(G)} \deg_G(u) + n) = O(m + n)$ time.

Then we compute the sets $L_{G,\sigma}^{\sigma^{-1}(u)-1}(u)$ and the numbers $|L_{G,\sigma}^{\sigma^{-1}(u)-1}(u)|$ for all $u \in V(G)$. This also can be done in $O(\sum_{u \in V(G)} \deg_G(u) + n) = O(m + n)$ time.

Let ϕ be an ordering of $V(G)$ such that $f_\sigma(\phi(1)) \geq f_\sigma(\phi(2)) \geq \cdots \geq f_\sigma(\phi(n))$. Since $1 \leq f_\sigma(u) \leq n$ for every $u \in V(G)$, we can compute ϕ in $O(n)$ time using, say counting sort [3, §8.2].

Similar to the linear implementation of LBFS* in [13, §7], we run three parallel LBFSs: one based on σ (say ρ_1), one on the reverse of σ (say ρ_2) and one on ϕ (say ρ_3). As in the algorithm description, we should put $\rho(1) = \sigma(n)$. We also set $\rho_1(1) = \rho_2(1) = \rho_3(1) = \sigma(n)$. Every time a slice $S_\rho(i), i \in [2, n]$, is encountered, the vertex $\sigma(\alpha')$ is the first vertex of $S_{\rho_1}(i)$ in σ, the vertex $\sigma(\beta')$ is the first vertex of $S_{\rho_2}(i)$ in the reverse of the ordering σ and the vertex w is the first vertex of $S_{\rho_3}(i)$ in the ordering ϕ. Note that the paradigm of "partitioning" allows us to pick $\sigma(\alpha'), \sigma(\beta')$ and w in the three parallel LBFSs, respectively, in $O(1)$ time. Note that after picking a vertex from a slice in a given LBFS, we do not necessarily visit this chosen vertex in the next step. Indeed, the Four-Sweep LBFS Recognition Algorithm will choose one of $\sigma(\alpha'), \sigma(\beta')$ and w as $\rho(i)$ and we will take $\rho_1(i) = \rho_2(i) = \rho_3(i) = \rho(i)$. Now, what is left to do is to implement efficiently the choice of $\rho(i)$. Let $u \in \{\sigma(\alpha'),\ \sigma(\beta'),\ w\}$ be a vertex which satisfies

$$\deg_G(u) = \min\{\deg_G(\sigma(\alpha')),\ \deg_G(\sigma(\beta')),\ \deg_G(w)\}.$$

We need $O(1)$ time to find u since the three numbers $\deg_G(\sigma(\alpha'))$, $\deg_G(\sigma(\beta'))$ and $\deg_G(w)$ have been calculated in advance.

CLAIM: Let $T = N_G(u) \cap \rho[i-1] \cap \sigma[\alpha'-1]$. If $|T| < |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$, then $L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] \neq \emptyset$. If $|T| = |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$, then $L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] = \emptyset$.

Proof: Since both u and $\sigma(\alpha')$ belong to $S_\rho(i)$, we get $N_G(\sigma(\alpha')) \cap \rho[i-1] = N_G(u) \cap \rho[i-1]$. Henceforth, we have

$$\begin{aligned} L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] &= L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus (N_G(\sigma(\alpha')) \cap \rho[i-1]) \\ &= L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus (N_G(\sigma(u)) \cap \rho[i-1]) \\ &= L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus T. \end{aligned} \tag{66}$$

If $|T| < |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$, then Eq. (66) gives $L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] \neq \emptyset$. If $|T| = |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$, as we surely have $T \subseteq L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))$, Eq. (66) leads to $L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] = \emptyset$, as was to be shown. □

After establishing the above claim, we continue illustrating the remaining steps to get $\rho(i)$ in $O(\deg_G(\rho(i)))$ $+O(1)$ time.

- Compute the set $T = N_G(u) \cap \rho[i-1] \cap \sigma[\alpha'-1]$ and its size $|T|$. Determine which is true, $|T| < |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$ or $|T| = |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$. Since $\deg_G(u) \leq \deg_G(\rho(i))$, this step can be done in $O(\deg_G(\rho(i)))$ time.

- If $|T| < |L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha'))|$, according to the Claim and our algorithm description, we should set $\rho(i)$ to be $\sigma(\alpha')$.

- If $L_{G,\sigma}^{\alpha'-1}(\sigma(\alpha')) \setminus \rho[i-1] = \emptyset$, according to the Claim and our algorithm description, we should proceed to check whether or not it holds $f_\sigma(w) > \beta'$ and this costs $O(1)$ time. If $f_\sigma(w) > \beta'$, we set $\rho(i) = w$ and else we set $\rho(i) = \sigma(\beta')$.

In all, we find that the algorithm LBFS$^\bigstar$ can be implemented in $O(\sum_{v\in V(G)} \deg_G(v) + n) = O(m + n)$ time. This also means that our Four-Sweep LBFS Recognition Algorithm can be implemented in linear time.

References

[1] P. BALDY, M. MORVAN, *A linear time and space algorithm to recognize interval orders*, Discrete Applied Mathematics, 46 (1993), pp. 173–178.

[2] K.S. BOOTH, G.S. LUEKER, *Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms*, Journal of Computer and System Sciences, 13 (1976), pp. 335–379.

[3] T.H. CORMAN, C.E. LEISERSON, R.L. RIVEST, C. STEIN, *Introduction to Algorithms*, Second Edition, The MIT Press, 2001.

[4] D.G. CORNEIL, *A simple 3-sweep* LBFS *algorithm for the recognition of unit interval graphs*, Discrete Applied Mathematics, 138 (2004), pp. 371–379.

[5] D.G. CORNEIL, *Lexicographic Breadth First Search – A survey*, Lecture Notes in Computer Science, 3353 (2005), pp. 1–19.

[6] D.G. CORNEIL, H. KIM, S. NATARAJAN, S. OLARIU, A.P. SPRAGUE, *Simple linear time recognition of unit interval graphs*, Information Processing Letters, 55 (1995), pp. 99–104.

[7] D.G. CORNEIL, E. KÖHLER, private communications.

[8] D.G. CORNEIL, E. KÖHLER, J-M. LANLIGNEL, *On end-vertices of Lexicographic Breadth First Searches*, Discrete Applied Mathematics, 158 (2010), pp. 434–443.

[9] D.G. CORNEIL, R.M. KRUEGER, *A unified view of graph searching*, SIAM Journal on Discrete Mathematics, 22 (2008), pp. 1259–1276.

[10] D.G. CORNEIL, S. OLARIU, L. STEWART, *Asteroidal triple-free graphs*, SIAM Journal on Discrete Mathematics, 10 (1997), pp. 399–430.

[11] D.G. CORNEIL, S. OLARIU, L. STEWART, *The ultimate interval graph recognition algorithm? (extended abstract),* Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, SIAM, Philadelphia, 1998, pp. 175–180.

[12] D.G. CORNEIL, S. OLARIU, L. STEWART, *Linear time algorithms for dominating pairs in asteroidal triple-free graphs*, SIAM Journal on Computing, 28 (1999), pp. 1284–1297.

[13] D.G. CORNEIL, S. OLARIU, L. STEWART, *The* LBFS *structure and recognition of interval graphs*, SIAM Journal on Discrete Mathematics, 23 (2009), pp. 1905–1953.

[14] C. CRESPELLE, *Fully dynamic representations of interval graphs*, Lecture Notes in Computer Science, 5911 (2010), pp. 77–87.

[15] X. DENG, P. HELL, J. HUANG, *Linear time representation algorithms for proper circular-arc graphs and proper interval graphs*, SIAM Journal on Computing, 25 (1996), pp. 390–403.

[16] F.F. DRAGAN, F. NICOLAI, A. BRANDSTÄDT, *LexBFS-orderings and powers of graphs*, Lecture Notes in Computer Science, 1197 (1997), pp. 166–180.

[17] C.M.H. DE FIGUEIREDO, J. MEIDANIS, C.P. DE MELLO, *A linear-time algorithm for proper interval graph recognition*, Information Processing Letters, 56 (1995), pp. 179–184.

[18] P.C. FISHBURN, *Interval Orders and Interval Graphs: A Study of Partially Ordered Sets*, John Wiley & Sons Inc., 1985.

[19] J. GIMBEL, *End vertices in interval graphs*, Discrete Applied Mathematics, 21 (1988), pp. 257–259.

[20] M.C. GOLUMBIC, *Algorithmic Graph Theory and Perfect Graphs*, 2nd ed., Annals of Discrete Mathematics, 57, Elsevier, Amsterdam, The Netherlands, 2004.

[21] M. HABIB, R.M. MCCONNELL, C. PAUL, L. VIENNOT, *Lex-BFS and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing*, Theoretical Computer Science, 234 (2000), pp. 59–84.

[22] P. HELL, J. HUANG, *Lexicographic orientation and representation algorithms for comparability graphs, proper circular arc graphs, and proper interval graphs*, Journal of Graph Theory, 20 (1995), pp. 361–374.

[23] P. HELL, J. HUANG, *Certifying LexBFS recognition algorithms for proper interval graphs and proper interval bigraphs*, SIAM Journal on Discrete Mathematics, 18 (2005), pp. 554–570.

[24] P. HELL, R. SHAMIR, R. SHARAN, *A fully dynamic algorithm for recognizing and representing proper interval graphs*, SIAM Journal on Computing, 31 (2001), pp. 289–305.

[25] W.-L. HSU, *A simple test for interval graphs*, Lecture Notes in Computer Science, 657 (1993), pp. 11–16.

[26] W.-L. HSU, $O(m \cdot n)$ *algorithms for the recognition and isomorphism problems on circular-arc graphs*, SIAM Journal on Computing, 24 (1995), pp. 411–439.

[27] W.-L. HSU, *On-line recognition of interval graphs in* $O(m + n \log n)$ *time*, Lecture Notes in Computer Science, 1120 (1996), pp. 27–38.

[28] W.-L. HSU, T-H. MA, *Fast and simple algorithms for recognizing chordal comparability graphs and interval graphs*, SIAM Journal on Computing, 28 (1999), pp. 1004–1020.

[29] L. IBARRA, *A fully dynamic graph algorithm for recognizing interval graphs*, Algorithmica, 58 (2010), pp. 637–678.

[30] L. IBARRA, *A fully dynamic graph algorithm for recognizing proper interval graphs*, Lecture Notes in Computer Science, 5431 (2009), pp. 190–201.

[31] B. JAMISON, S. OLARIU, *On the semi-perfect elimination,* Advances in Applied Mathematics, 9 (1988), pp. 364–376.

[32] N. KORTE, R.H. MÖHRING, *An incremental linear-time algorithm for recognizing interval graphs*, SIAM Journal on Computing, 18 (1989), pp. 68–81.

[33] D. KRATSCH, R.M. MCCONNELL, K. MEHLHORN, J.P. SPINRAD, *Certifying algorithms for recognizing interval graphs and permutation graphs*, SIAM Journal on Computing, 36 (2006), pp. 326–353.

[34] C.G. LEKKERKERKER, J.C. BOLAND, *Representation of a finite graph by a set of intervals on the real line*, Fundamenta Mathematicae, 51 (1962), pp. 45–64.

[35] P. LI, Y. WU, *Maximal neighborhood search and rigid interval graphs*, Journal of Graph Algorithms and Applications, 17 (2013), pp. 245–264.

[36] P. LI, Y. WU, *A 4-sweep* LBFS *certifying algorithm for recognizing interval graphs*, in preparation.

[37] P.J. LOOGES, S. OLARIU, *Optimal greedy algorithms for indifference graphs*, Computers & Mathematics with Applications, 25 (1993), pp. 15–25.

[38] W-F. LU, W-L. HSU, *A clustering algorithm for interval graph test on noisy data*, Lecture Notes in Computer Science, 2647 (2003), pp. 195–208.

[39] T. MA, unpublished manuscript.

[40] D. MEISTER, *Recognition and computation of minimal triangulations for AT-free claw-free and co-comparability graphs*, Discrete Applied Mathematics, 146 (2005), pp. 193–218.

[41] R.H. MÖHRING, *Algorithmic aspects of comparability graphs and interval graphs*, in: I. Rival, editor, Graphs and Orders, pp. 41–101, D. Reidel, Boston, 1985.

[42] B.S. PANDA, S.K. DAS, *A linear time recognition algorithm for proper interval graphs*, Information Processing Letters, 87 (2003), pp. 153–161.

[43] G. RAMALINGAM, C.P. RANGAN, *A uniform approach to domination problems on interval graphs*, Information Processing Letters, 27 (1988), pp. 271–274.

[44] A. RAYCHAUDHURI, *On powers of interval and unit interval graphs*, Congressus Numerantium, 59 (1987), pp. 235–242.

[45] F.S. ROBERTS, *Indifference graphs*, F. Harary (Ed.), Proof Techniques in Graph Theory, Academic Press, New York, (1969), pp. 139–146.

[46] D.J. ROSE, R.E. TARJAN, G.S. LUEKER, *Algorithmic aspects of vertex elimination on graphs*, SIAM Journal on Computing, 5 (1976), pp. 266–283.

[47] K. SIMON, *A new simple linear algorithm to recognize interval graphs*, Lecture Notes in Computer Science, 553 (1992), pp. 289–308.

[48] J.P. SPINRAD, *Efficient Graph Representations*, Fields Institute Monographs, 19, American Mathematical Society, 2003.

[49] W.T. TROTTER, *New perspectives on interval orders and interval graphs*, in: R.A. Bailey, Ed., Surveys in Combinatorics, pp. 237–286, London Mathematical Society Lecture Note Series 241, Cambridge University Press, Cambridge, 1997.